

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Факультет електроніки
(повна назва інституту/факультету)

Кафедра мікроелектроніки
(повна назва кафедри)

«На правах рукопису»
УДК _____

«До захисту допущено»

В. о. завідувача кафедри
_____ Анатолій ОРЛОВ
(підпис) (ім'я, ПРІЗВИЩЕ)

“___” грудня 2020 р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 153 – Мікро та наносистемна техніка
(код і назва)

на тему: Ультразвуковий розпилювач рідини для інгаляції та контролю вологості

Виконав (-ла): студент 2 курсу, групи ДП-92мп
(шифр групи)

Юрчик Юрій Іванович
(прізвище, ім'я, по батькові)

_____ (підпис)

Науковий керівник проф., д.т.н. Якименко Ю.І.
(посада, науковий ступінь, вчене звання, прізвище та ініціали)

_____ (підпис)

Консультант з нормоконтролю доц., к.ф.-м.н., с.н.с. Свечніков Г.С.

_____ (підпис)

Консультант з інформаційних питань доц., к.т.н. Діденко Ю.В.

_____ (підпис)

Рецензент доц. каф. ЕІ, к.т.н., доц. Казміренко В.А.
(посада, науковий ступінь, вчене звання, прізвище та ініціали)

_____ (підпис)

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць інших
авторів без відповідних посилань.
Студент

Київ – 2020

Реферат

У цій роботі описується процес розробки цифрового зволожувача з використанням режиму інгаляції та представлені результати виконаної роботи, а саме принципова схема та програмний код пристрою. Процес розробки поділяється на кілька етапів. Наведена інформація про вплив вологості та дані про актуальність створення пристроїв, що підвищують вологість, а також дані про ефективність парової інгаляції при лікуванні респіраторних захворювань. На основі цих даних вибирається загальний перелік компонентів, необхідних для реалізації пристрою.

Для основних компонентів пристрою розглядаються різні варіанти реалізації та описуються загальні принципи роботи, переваги та недоліки. На основі опису компонентів наводяться переваги та недоліки кожної реалізації компонентів у розробленому пристрої. На основі цих даних вибрана така реалізація, яка найбільш ефективно відповідає вимогам.

Розроблено прототип обладнання та програмного забезпечення для управління ним. Платформа Arduino використана для взаємодії між підключеними компонентами.

Abstract

This paper describes the process of developing a digital humidifier using the inhalation mode and presents the results of the work performed, namely the schematic diagram and program code of the device. The development process is divided into several stages. Information on the influence of humidity and data on the relevance of creating devices that increase humidity, as well as data on the effectiveness of steam inhalation in the treatment of respiratory diseases. Based on these data, a general list of components required for the implementation of the device is selected.

For the main components of the device, various implementation options are considered and the general principles of operation, advantages and disadvantages are described. Based on the description of the components, the advantages and disadvantages of each implementation of the components in the developed device are given. Based on these data, the implementation that most effectively meets the requirements is selected.

A prototype of hardware and software for its management has been developed. The Arduino platform is used for interaction between connected components.

Зміст

1.	Вступ.....	7
1.1.	Значення вологості.....	7
1.2.	Вологість в електронних виробах	9
1.3.	Співвідношення між інгаляцією та лікуванням респіраторних захворювань	9
1.4.	Формулювання задачі.....	10
1.5.	Висновок розділу	11
2.	Варіанти реалізації ключових одиниць обладнання.....	12
2.1.	Датчик вимірювання температури	12
2.1.1.	Піроелектричний датчик.....	12
2.1.2.	Терморезистор	13
2.1.3.	Кварцові датчики.....	16
2.2.	Зволожувач та технологія зволоження	17
2.2.1.	Гарячий випарник.....	18
2.2.2.	Холодний випарник.....	18
2.3.	Датчик вологості	20
2.3.1.	Ємнісний датчик.....	20
2.3.2.	Гігрометр опору.....	21
2.4.	Мікроконтролер для керування пристроєм.....	21
2.4.1.	Мікроконтролери Atmel.....	21
2.4.2.	Мікроконтролери PIC.....	25
2.4.3.	Мікроконтролери STM32.....	29
2.5.	Висновок розділу	31
3.	Технології та компоненти, що використовуються для реалізації пристрою	32

3.1.	Вибір мікроконтролера для приладу.....	32
3.2.	Вибір екрану для виведення інформації.....	34
3.3.	Вибір датчика температури для виконання завдання	36
3.4.	Вибір випарника.....	37
3.5.	Вибір датчика вологості	38
3.6.	Вибір вентилятора.....	38
3.7.	ШИМ.....	39
3.7.1.	ШИМ-регулювання потужністю	39
3.7.1.	Робота з ШИМ в Arduino.....	40
3.8.	Висновок розділу	41
4.	Схема роботи приладу та керуючий код	42
4.1.	Програмування в Arduino.....	42
4.1.1.	Цифрові входи.....	42
4.1.2.	Аналогові входи.....	43
4.1.3.	Функції setup() і loop().....	43
4.2.	Робота з датчиком вологості.....	43
4.3.	Робота з датчиком температури	45
4.4.	Кнопки керування	48
4.5.	Використання дисплею	50
4.5.1.	Фізичний зв'язок дисплею.....	50
4.5.2.	Програмне підключення індикатора для відображення та виведення інформації	54
4.5.3.	Символ градуса для дисплею	57
4.5.4.	Налаштування підсвічування	58
4.6.	Підключення та управління випарником та вентилятором.....	59

4.7.	Меню пристрою та код управління.....	61
4.8.	Розробка живлення	64
4.1.	Висновок до поточного розділу	66
5.	Стартап-проект	68
5.1.	Назва проекту	68
5.2.	Короткий опис проекту	68
5.3.	Бізнес-модель	68
5.3.1.	Цінний продукт.....	68
5.3.2.	Сегмент споживачів	69
5.3.3.	Канали збуту	69
5.3.4.	Взаємодія зі споживачами	69
5.3.5.	Прибуток (монетизація).....	70
5.3.6.	Ключові види діяльності.....	70
5.3.7.	Ключові ресурси	70
5.3.8.	Ключові партнери.....	70
5.3.9.	Витрати	71
5.3.10.	Споживчі властивості товару	71
5.3.11.	Дослідження ринку.....	71
5.3.12.	Дослідження конкурентного оточення.....	72
5.3.13.	Маркетингова стратегія просування	72
5.4.	Елементи фінансового плану	72
5.4.1.	Опис бізнес-проекту.....	72
6.	Висновки	73
7.	Перелік використаної літератури	74

1. Вступ

Кількість водяної пари, яка присутня в повітрі, впливає не тільки на особистий комфорт, але також впливає на різні виробничі процеси в промислових цілях. Наприклад, у напівпровідниковій промисловості рівень вологості повинен контролюватися, щоб забезпечити належне поводження з пластинами. Контроль вологості також важливий для інкубаторів, дихального обладнання, стерилізаторів та біопрепаратів. Крім того, наявність водяної пари впливатиме на різні хімічні, біологічні та фізичні процеси. У промисловості вимірювання вологості часто є важливим, оскільки це впливає на здоров'я та безпеку персоналу та вартість продукції. Вологість є важливою частиною прогнозування погоди, оскільки вона вказує на можливість конденсації, туману або опадів. Більш висока відносна вологість зменшить ефективність охолодження кузова автомобіля. Це пов'язано з тим, що випаровування зі шкіри уповільнюється. Як результат, влітку однакова температура відчувається вищою в місці з високою вологістю.

1.1. Значення вологості

Кількість водяної пари, необхідної для досягнення насичення, зростає із збільшенням температури. Зі зниженням температури повітря вона досягає точки насичення, не збільшуючи і не зменшуючи кількість води. Кількість водяної пари, що міститься в повітрі, може сильно відрізнитися.

Існує два типи записів вологості: абсолютна і відносна вологість. Абсолютна вологість характеризує вміст вологи в повітрі, виражену в грамах/м³ або грамах/кг [1]. Відносна вологість, виражена у відсотках, являє собою поточний стан абсолютної відносної вологості при тій же температурі.

Абсолютна вологість - це загальна маса присутності водяної пари в одиниці об'єму повітря. У цьому випадку температура не враховується. Коли

повітря насичене при 30°C (86°F), абсолютна вологість повітря в атмосфері становить від нуля до приблизно 30 г/м^3 [2].

Коли температура або тиск повітря змінюється при нефіксованій величині об'єму, змінюється абсолютна вологість. Це робить її непридатним для хімічних інженерних розрахунків, наприклад, під час сушіння температура може сильно відрізнятись. Отже, абсолютна вологість повітря визначається діленням кількості водяної пари, що міститься в даному об'ємі повітря при певній температурі, на об'єм повітря. Значення вказано в одиницях грам/м^3 або грам/кг . Ця одиниця виміру підходить для розрахунків та практичного застосування [3].

Відносна вологість повітря визначається як співвідношення наявної в повітрі водяної пари до максимального об'єму водяної пари, який може утримати повітря при даній температурі в незмінному об'ємі.

За рівнем відносної вологості створюють прогнози погоди та звіти щодо ймовірності різних видів опадів у тому числі ймовірність випадання роси чи утворення туману. У спекотне літо збільшення відносної вологості збільшить температуру повітря, яку відчують люди і тварини, тим самим запобігаючи випаровуванню поту з шкіри.

Вологість грає важливу роль у житті організмів. Для тварин, які використовують піт для регулювання температури тіла, висока вологість зменшить швидкість випаровування води на поверхні шкіри, зменшуючи тим самим ефективність тепловіддачі.

Людське тіло використовує випаровування води в якості головного механізму терморегуляції, тому люди чутливі до змін вологості повітря. За високої вологості повітря швидкість випаровування води з поверхонь, в тому числі зі шкіри людини, нижча порівняно з сухими умовами. За високої вологості температура повітря здається вищою, бо процес випаровування зі шкіри сповільнюється і тіло охолоджується повільніше, ніж за низької вологості.

Кондиціонер зменшує дискомфорт, не тільки знижує температуру, але й знижує вологість. Нагрівання зовнішнього холодного повітря може зменшити відносну вологість приміщення нижче 30% [4], що може спричинити такі незручності, як сухість шкіри, потріскані губи, сухість очей та спрага.

1.2. Вологість в електронних виробках

Електронні пристрої зазвичай призначені для роботи лише за певних умов вологості (наприклад, від 5% до 95%). У найвищій точці в цьому діапазоні вологість збільшить провідність ізолятора, спричиняючи руйнування. Низька вологість робить матеріал крихким. Цей рівень вологості може спричинити накопичення статичної електрики, що може призвести до самовільного вимкнення комп'ютера при розряді. Окрім помилкової нестабільності, електростатичний розряд може також спричинити діелектричний пробій у твердотільних пристроях, що може призвести до незворотних пошкоджень. З цих причин центри обробки даних часто контролюють рівень відносної вологості [5].

1.3. Співвідношення між інгаляцією та лікуванням респіраторних захворювань

Для лікування гострих респіраторних захворювань популярні теплові інгаляції. Для цього використовуйте спеціальний паровий інгалятор та побутові пристрої. Однак ефективність та терапевтична ефективність цієї інгаляції дуже низька. На сьогоднішній день люди знають багато про розмір частинок, що утворюються за технологією інгаляції, і як їх потрібно подрібнити, щоб препарат дійшов до потрібного органу та мав терапевтичний ефект. Тому частинки діаметром більше 10 мкм осідають в ротоглотці. 5 - 10 мкм - у глотці, гортані та трахеї; від 1 до 5 мкм - у нижніх дихальних шляхах;

0,5 - 1 мкм - досягають альвеол; менше 0,5 мкм - не потрапляють на органи і не виходять під час видиху.

До переваг інгаляційної терапії належать: висока концентрація ліків, що створюється в легенях, відсутність біотрансформації ліків (вони не зв'язуються з білками крові, не модифікується в печінці тощо) для досягнення терапевтичних ефектів. Зменшує негативний вплив певних препаратів (антибіотиків) на інші органи людини [6].

Інгаляції широко застосовуються при лікуванні обструктивних захворювань. Небулайзери застосовують при захворюваннях та важкій бронхіальній астмі та інших хронічних обструктивних захворюваннях легень (професійні захворювання легень, бронхіти, бронхоектатичні захворювання, хронічний обструктивний бронхіт та ін.). Зазвичай використання небулайзера - єдиний спосіб доставити певні ліки (муколітики, антибіотики) до уражених органів. Без використання інгаляційної терапії лікування дітей до року та хворих та важких хворих, як правило, ускладнене [6].

1.4. Формулювання задачі

У цій роботі необхідно розробити пристрій, який поєднає в собі автоматичний зволожувач повітря та інгалятор. Для виконання функції автоматичного зволоження пристрій повинен бути обладнаний датчиком вологості, а випарник повинен бути розміщений так, щоб потік пари не впливав суттєво на поточні показники, інакше зволоження зупинятиметься кожні кілька хвилин, поки поблизу датчика не знизиться вологість. Для визначення температури рідини для вдихання необхідно використовувати датчик температури. Відповідно до сигналу датчика, роботу буде зупинено при низькій або високій температурі. Панель управління повинна мати кнопки та екран, щоб вказувати поточний режим, поточний рівень вологості та регулювати відносну вологість, яку потрібно досягти. В режимі зволоження прилад має швидко збільшити вологість у приміщенні, тому він повинен

працювати на високій потужності. Метод інгаляції повинен передбачати регулювання тривалості інгаляції та інтенсивності випаровування води (фармацевтичні препарати). Поточні дані повинні відображатися на екрані в режимі всмоктування: потужність випаровування у відсотках, час від початку до кінця інгаляції, поточна температура рідини в контейнері (у градусах Цельсія). Основним компонентом пристрою повинен бути контролер Arduino. Тому в наступних розділах необхідно вибрати відповідне обладнання для випаровування; датчики для вимірювання температури та вологості води; контролер Arduino, який може відповідати задачі та повною мірою використовувати свої функції; забезпечити регулювання потужності випаровування; вибрати екран індикації; вибрати вентилятор для виносу зволоженого повітря із контейнера; розробити живлення контролера, датчика, індикатора, випарника та вентилятора.

1.5. Висновок розділу

З метою контролю вологості в будинку вона вимірюється у відносних одиницях. Представлена роль вологості у підтримці здоров'я людини в природі та її роль в побутових приладах (тобто електронних виробках). Наведена актуальність використання інгаляційної терапії для лікування дихальних шляхів людини. Крім того, він також перелічені компоненти, необхідні для реалізації необхідних функцій, які потрібно розробити у наступних розділах.

2. Варіанти реалізації ключових одиниць обладнання

2.1. Датчик вимірювання температури

На даний момент існує кілька типів датчиків, які можуть вимірювати температуру:

- термоелектричний (термопара);
- терморезистивний;
- п'єзоелектричний.
- пірометричний;
- акустичний.

Залежно від завдання потрібно виміряти температуру води, тому датчик повинен працювати в діапазоні від 0 до 100°C. Датчики, засновані на безконтактних методах вимірювання температури, не слід використовувати в цьому діапазоні, оскільки вони набагато дорожчі за контактні і ефективні для використання лише тоді, коли контакт використовувати не можна або дані про температуру в режимі реального часу потрібно отримувати з мінімальною затримкою. У даному випадку висока точність та низька затримка вимірювання не потрібні.

2.1.1. Піроелектричний датчик

Термоелектричні термометри використовують явище термоелектричної рушійної сили (термоЕРС) у точці контакту двох металів або сплавів, яке залежить від температури переходу на обох кінцях двох різних провідників (чутливих компонентів термопар). Відповідно до мінливого закону термоЕРС та вимірювання його розміру за допомогою вимірювального пристрою можна розрахувати температуру місця з'єднання.

Термоелектричний термометр має два термоелектроди, з'єднані на одному кінці і ізольовані в інших, вони мають захисний кожух і головку з фіксатором, через які датчик підключений до вимірювальної лінії. Це

первинний перетворювач, тому його слід використовувати з вторинним перетворювачем, який перетворює напругу в температуру. Вторинним перетворювачем може бути мілівольтметр, цифровий пристрій.

Термопари дозволяють вимірювати опір в діапазоні від -200 до 2500 $^{\circ}\text{C}$. Основне поле, яке вони використовують, - це вимірювання високих температур, при яких важко або неможливо використовувати інші датчики [9].

2.1.2. Терморезистор

Терморезистор - це пристрій, заснований на зміні опору речовини з температурою. Використовуються при вимірюванні температури від -260 до 750 $^{\circ}\text{C}$. Певні датчики можна використовувати для вимірювання температури до 1000 $^{\circ}\text{C}$. Більшість металів мають позитивний температурний коефіцієнт опору, тобто їх опір зростає при нагріванні. Це пов'язано з посиленням розсіюванням електронів на неоднорідності решітки, що зумовлене посиленням тепловим коливанням іона поблизу його положення рівноваги. У напівпровідниках при нагріванні число електронів провідності зростає із збільшенням температури.

Основним компонентом є терморезистор - компонент, який змінює свій опір відповідно до температури навколишнього середовища. На рисунку показано кілька прикладів зовнішнього вигляду датчиків теплового опору. [10]

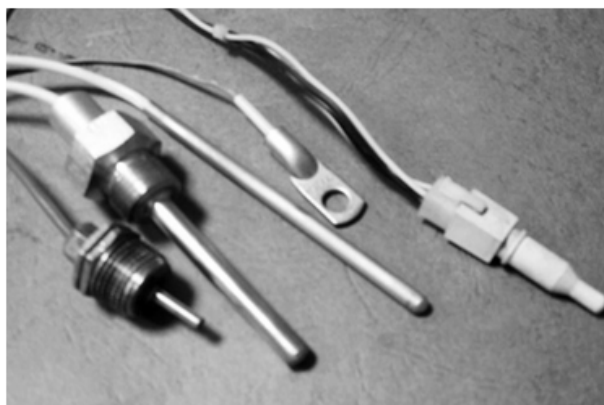


Рис. 2.1 – Приклади терморезистивних датчиків

Перевагами датчиків теплового опору є висока чутливість, прості у створенні схеми вимірювання та тривала стабільність. Такі датчики можна розділити на дві підгрупи: датчики температурного опору і термістор. [11]

Оскільки питомий опір всіх металів і більшості сплавів залежить від температури, їх можна використовувати для розробки чутливих компонентів для вимірювання температури. Для виготовлення датчиків температури придатні майже всі метали, але в більшості випадків для цих цілей використовується тільки платина. Це обумовлено відтворюваністю, довготривалою стабільністю та довговічністю. Вольфрамові датчики використовуються для вимірювання температури вище 600 °C. Всі металеві датчики мають позитивний температурний коефіцієнт. Існує кілька типів металевих датчиків:

1. Тонкоплівкові датчики, виготовлені з платини або тонких шарів її сплавів, наносяться на відповідні основи, такі як мікромембрани з кремнію. Їй зазвичай наносять у формі змієподібної структури для отримання високого співвідношення сторін.

2. Дротяні датчики, в яких платиновий дріт намотується на керамічну трубку і кріпиться до неї за допомогою високотемпературного клею. Така конструкція дозволяє виготовляти датчики з високою стабільністю. [11]

Терморезистор - напівпровідниковий пристрій, опір якого змінюється відповідно до його температури. Термістори виготовляються з матеріалів з високим температурним коефіцієнтом опору (ТКО), які, як правило, на порядок вищі, ніж ТКО металів та сплавів металів. [11]

Температурний коефіцієнт опору - значення рівне відносній зміні опору матеріалу на одиницю температури. Температурний коефіцієнт опору являє собою залежність опору від температури, виражену в кельвінах у мінус першому степені. Для більшості металів температурний коефіцієнт опору позитивний: внаслідок розсіювання електронів фононами (теплові коливання решітки) опір зростає із збільшенням температури.

Для напівпровідників без домішок він є негативним (опір зменшується із збільшенням температури), оскільки зі збільшенням температури все більше і більше електронів потрапляє в зону провідності, і концентрація дірок відповідно збільшується. Температурна залежність опору металевих сплавів, газів, легованих напівпровідників та електролітів є більш складною. [11]

Основне застосування терморезистора - вимірювання температури. Температура є важливим параметром для роботи різного побутового та промислового обладнання. Так само значення температури має великий вплив на різні процеси, особливо в хімічній промисловості. Відповідно до умов вимірювання для вимірювання використовуються різні датчики температури, включаючи терморезистивні.

Термістор є пасивною складовою ланцюга, і його опір залежить від температури. Тому для вимірювання температури можна використовувати будь-який метод вимірювання опору.

При нагріванні це зменшує опір напівпровідника. Абсолютне значення температурного коефіцієнта опору в напівпровідниках вище, ніж у чистих металах. Вимірювання, проведене за допомогою термометра опору, має відбуватись при безпосередньому контакті з вимірюваним середовищем [9]. Відповідно до діаграми характеристики термометра можна визначити температуру навколишнього середовища, що контактує з ним. Переваги металевих термометрів опору полягають у наступному:

- можливість виготовлення обладнання практично в будь-якому температурному діапазоні;
- висока точність вимірювання;
- безліч датчиків можна підключити до одного вимірювального пристрою через комутаційну систему;
- можливість використовувати його з цифровим обладнанням.

Основним недоліком цього виду термометра є те, що його потрібно підключати до джерела постійного струму, що може спотворити результат

вимірювання за відносно низького опору датчика та високої напруги живлення.

Чутливість напівпровідникових термісторів набагато вища, ніж у металів, що є їх основною перевагою, але це також обмежує діапазон вимірюваних температур. Це пов'язано з тим, що опір напівпровідникового термістора змінюється в десятки і навіть тисячі разів в межах вимірювального діапазону, що створює труднощі в проектній схемі вимірювального приладу. Напівпровідникові термометри дозволяють вимірювати температури в діапазоні від -80 до $150\text{ }^{\circ}\text{C}$ [13].

2.1.3. Кварцові датчики

Кварцовий термометр - це термометр, заснований на температурній залежності резонансної частоти п'єзоелемента. Частота п'єзоелектричного кристала має високу стабільність. Кварцові термометри не підходять для вимірювань, що вимагають високої точності. Це пов'язано з його гістерезисом. Однак його дрейф з часом дуже низький, менше $0,005\text{ K}$ на місяць. Такий термометр має високу температурну роздільну здатність, тому його можна використовувати для відносного вимірювання з високою точністю. Крім того, наявність частотних вихідних параметрів дає можливість використовувати в автоматичних ланцюгах, які самостійно регулюють тепловий стан об'єктів [13].

Кварцовий датчик використовує п'єзоелектричний резонатор, як чутливий елемент частота якого сильно залежить від температури. Перш за все, перевагами використання термо кварцових резонаторів є їх висока чутливість, висока стабільність та простота використання. Сигнал від резонатора може бути негайно оброблений у цифровій формі, зменшуючи тим самим витрати на процес регулювання температури. Вимірювання температури за допомогою теплового кварцового резонатора базується на використанні анізотропії кристалів кварцу.

Чутливість сучасних кварцових терморезонаторів становить $60 \text{ ppm}/^{\circ}\text{C}$. Для резонаторів з опорними частотами 32 кГц і 64 кГц вони становлять $2 \text{ Гц}/^{\circ}\text{C}$ і $4 \text{ Гц}/^{\circ}\text{C}$. Діапазон вимірюваних температур обмежується температурою азоту внизу та приблизно $+(150 \dots 200)^{\circ}\text{C}$ вгорі. Це обмеження залежить від того, чи існують провали в резонаторі при використанні штучної кварцової сировини, і зменшенням крутизни характеристики при зниженні температури. Для досягнення високоточного вимірювання температури потрібно окреме калібрування та врахування нелінійного доданку у наближеному поліномі, однак це наразі не є складним завданням. Мікропроцесор перетворює значення частоти кварцового перетворювача у значення температури відповідно до кожної індексної характеристики. Існують термометри, які дозволяють вимірювати температуру з точністю до $0,06^{\circ}\text{C}$ в діапазоні $-30 \dots +100^{\circ}\text{C}$.

Типовий кварцовий термометр складається з трьох основних компонентів: чутливого елемента, перетворювача частоти та спеціального калькулятора (мікроконтролера). Цифрові термометри на основі кварцових резонаторів можуть використовуватися в якості багатоканальних систем регулювання температури. Можливо передавати частоти на декілька сотень метрів від перетворювача температури. Основне призначення кварцового термометра - це дослідження, пов'язані з високою точністю та тривалим вимірюванням. [12]

2.2. Зволожувач та технологія зволоження

Нижче наведено найпоширеніші типи випарників води:

- гарячий випарник пари;
- холодної випарник пари;
- ультразвуковий випарник;
- компресорний випарник.

2.2.1. Гарячий випарник

Принцип роботи цих випарників заснований на тепловому випаровуванні води, що є звичайним процесом кипіння води, в результаті чого вода перетворюється на пару і насичує вологу в повітрі. Крім того, зволоження проводиться стерилізованою парою. У зволожувачі повітря схема роботи дуже схожа на схему роботи електричного чайника: електроди отримують енергію, вода нагрівається, і вона починає кипіти і випаровуватися. Коли вода повністю випаровується, ланцюг відключається, і пристрій автоматично відключається. Безпека експлуатації закладена в конструкції зволожувача. З точки зору пожежної безпеки, цей зволожувач повітря не небезпечніший за звичайні побутові чайники.

Паровий зволожувач має просту конструкцію і відносно низьку ціну. Але їх низькі ціни значною мірою компенсуються високим споживанням енергії. Продуктивність таких пристроїв дуже висока - вони можуть за короткий час підвищити вологість до 80-100%. Шум, викликаний окропом, мінімальний, і приміщення опалюється гарячою парою. Однак, якщо температура повітря в будинку висока, надлишок тепла можна розцінити як недолік [12].

2.2.2. Холодний випарник

2.2.2.1. Ультразвуковий випарник

Ультразвукові випарники в даний час є найбільш ефективними. Вода з резервуара для води направляється на плоску пластину, яка вібрує з високою (ультразвуковою) частотою, де вода розпадається на дрібні частинки. Повітряний потік, який генерується вентилятором, надсилає його в кімнату і перетворюється у стан пари в кімнаті.

Тому ультразвуковий випарник дозволяє утворювати вдома туман. Пара від пристрою виглядає гарячою, але є насправді холодною, що абсолютно безпечно для здоров'я людей і рослин.

На відміну від парових випарників, перевагами ультразвуку є точне регулювання вологості, нижча температура пари джерела (залежно від температури води) та нижчий рівень шуму. В ультразвуковому випарнику рекомендується використовувати деіонізовану або дистильовану воду. Щоденна потужність ультразвукового випарника становить 7-12 літрів, а енергоспоживання менше 40-50 Вт [15].

Ультразвукові випарники використовують енергію високочастотних п'єзоелектричних коливань кристалів, що робить їх роботу майже безшумною і скорочує час всмоктування. Вони працюють безшумно, оскільки п'єзоелектричний кристал коливається в межах декількох сотень кілогерц, що вище частоти, яку може сприймати людське вухо. Більше 90% рідини, яка потрапляє в нижні дихальні шляхи, має середній розмір частинок від 2 до 3 мкм. Це робить їх доступними для інгаляцій.

Оскільки цей аерозоль досягає бронхіол та бронхіол з більшою концентрацією, терапевтичний ефект посилюється. Залишкова кількість препарату після інгаляції становить менше 0,5 мл, що дозволяє розпорошувати препарат з мінімальними втратами [6].

2.2.2.2. Компресорний випарник

Компресорний випарник розкладає рідину, яка повинна випаровуватися, на дрібні частинки сильним потоком повітря. Цей принцип розпилення генерує високий рівень шуму під час роботи та призводить до більших кінцевих розмірів пристрою. Цей випарник не можна ефективно використовувати у зволожувачах повітря у приміщенні, оскільки вони викликають значний дискомфорт під час безперервної роботи через вищезазначені недоліки. Їх використання стосується лише спреїв для інгаляцій.

2.3. Датчик вологості

Датчики вологості поділяються на:

- ємнісні;
- резистивні;

2.3.1. Ємнісний датчик

У простому варіанті реалізації датчиком вологості, заснованим на зміні ємності, є конденсатор з повітряним діелектриком між пластинами. Діелектрична проникність залежить від вологості, а потім змінюється вологість, що призводить до зміни ємності конденсатора.

Більш складна версія ємнісного гігрометра - це датчик з діелектриком, діелектрична проникність якого значною мірою залежить від поточної вологості. Матеріал, що перевіряється, розміщений між пластинами конденсатора, а сам конденсатор підключений до коливального контуру з генератором частоти. Спочатку вимірюють частоту власних коливань ланцюга, а потім вимірюють частоту з матеріалом. Після введення зразка для дослідження вимірювана частота визначає ємність датчика.

До переваг датчиків поточного типу належать лінійна залежність ємності від вологості, низька інерційність, низька вартість і висока точність вимірювання.

Ємнісні датчики мають певні недоліки. Вимірювана ними вологість становить менше 0,5% і має очевидну похибку, а випробовуваний зразок не повинен містити матеріалів з високою діелектричною постійною, оскільки це серйозно спотворює результати вимірювань. Також важливо, щоб досліджуваний зразок залишався незмінним [13].

2.3.2. Гігрометр опору

Конструктивно датчик опору складається з двох електродів на підкладці. Зверху електрод покритий шаром матеріалу з низьким опором, який значно змінюється при вологості. Глинозем найчастіше використовується через його дешевизну та популярність. Цей матеріал може поглинати відносно велику кількість води, що призводить до значної зміни опору датчика. Опір можна визначити за законом Ома шляхом вимірювання прикладеної напруги та струму через датчик.

Важливими перевагами резистивних датчиків є їх низька вартість, проста конструкція та логарифмічна залежність опору від вологості, що спрощує перетворення опір-вологість, малі розміри та стабільність характеристик у часі [16].

2.4. Мікроконтролер для керування пристроєм

2.4.1. Мікроконтролери Atmel

Atmel – провідних виробних мікросхем з енергонезалежною пам'яттю, мікроконтролерів та логічних мікросхем з можливістю їх програмування. Вона розробляє мікроконтролери на RISC архітектурі протягом 25 років і використовує всі розроблені рішення. Ідея нового швидкісного мікроконтролера створена розробниками в Норвегії в Дослідницькому центрі ATMEL. Перші мікроконтролери AVR AT90S вийшли в середині 1998 року і швидко завоювали популярність. [17]

Представлена мікроконтролерами серії AT90S архітектура AVR, поєднує в собі високошвидкісний процесор RISC, який отримує доступ до пам'яті програм та даних, вдосконалену систему команд, 32 регістри для загального призначення (при цьому кожен може бути регістром акумуляторів).

Більшість команд виконуються в одному машинному циклі, виконуючи поточну команду процесор вибирає наступну команду, що може забезпечити продуктивність до 1 MIPS/МГц тактової частоти.

Гарвардська архітектура AVR не тільки реалізує повне логічне та фізичне розділення адресного простору, але також реалізує повне логічне та фізичне розділення інформаційної шини для доступу до пам'яті програми та пам'яті даних. Спосіб вирішення та доступ до них також різний. Ця структура наближається до структури високошвидкісного цифрового процесора сигналів і значно покращила продуктивність завдяки наступним причинам:

а) процесор працює одночасно, включаючи пам'ять програм та пам'ять даних;

б) збільшення до 16 бітів шини даних пам'яті програм. [18]

З регістрів загального призначення в мікроконтролері утворений регістр швидкого доступу, який має безпосередній зв'язок з АЛП. За один цикл АЛП вибирає пару операндів з регістру, проводить обчислення та повертає результат в регістр. АЛП може проводити логічні, арифметичні операції з регістрами, між регістрами та константами. Крім того регістр може бути використаний як додаткова пам'ять даних. Шість регістрів з наявних 32 можуть бути використані в якості трьох 16-бітових регістрів-показчиків для непрямої адресації. В минулому с AVR мікроконтролери були з вбудованим в АЛП множитком, крім того стандартно в AVR налічується 120 команд. У всіх AVR мікроконтролерах присутня перепрограмована пам'ять, в яку можна записати код в готовій схемі, використовуючи інтерфейс з 4 ліній. [17]

До периферійних пристроїв AVR-мікроконтролерів належать: таймер, 10-бітний 8-канальний аналого-цифровий перетворювач (АЦП), підтримка зовнішніх переривань, широтно-імпульсний модулятор, пристрій скидання при увімкненні живлення, аналоговий компаратор, паралельний порт (від 3 до 48 вхідних та вихідних ліній), сторожовий таймер, інтерфейс UART та SPI. Усі ці функції роблять мікроконтролер AVR потужним інструментом для створення сучасних, високопродуктивних та економічних контролерів різного

призначення. Мікроконтролер AVR підтримує режим сну та режим енергоспоживання. Залежно від моделі мікроконтролер AVR працює в діапазоні напруг 2,7-6 В або 4-6 В. [17]

Переваги:

- високе співвідношення швидкодії до споживання енергії;
- зручність програмування;
- велика кількість моделей;
- широке поширення програмно-апаратних засобів для роботи з ними;
- висока навантажувальна здатність портів.

Області застосування AVR дуже широкі. Зокрема автомобільні датчики різного призначення з обчислювальним інтерфейсом, іграшки, ігрові приставки, материнські плати ПК, контролери для мобільних телефонів, зарядні пристрої, детектори диму та полум'я, побутова техніка. Сімейство невеликих AVR "tiny" знайшли використання в різних модемах, розумних зарядних пристроях, виробках класу Smart Card і пристроях для роботи з ними, супутниковій навігації для визначення координат автомобілів на трасі, складній побутовій техніці, пультах для дистанційного керування, мережевих картах комп'ютерів, материнських платах комп'ютерів, мобільних телефонах нового покоління, а також різних і різноманітних промислових системах для контролю та управління. AVR лінійки "mega" знайшли використання в аналогових і цифрових мобільних телефонах, принтерах і контролерах для них, контролерах факсових апаратів, контролерах сучасних твердотільних дисків і т.д. [17]

На основі мікроконтролерів Atmel представлена лінійка плат для розробки Arduino. Arduino та його аналоги - це набори, що складаються з готових електронних блоків та програмного забезпечення. У цьому випадку блок електронного обладнання - це друкована плата, на якій встановлений мікроконтролер, і для його роботи потрібні мінімальні електронні компоненти. Насправді електронний блок Arduino схожий на материнську плату сучасного комп'ютера. Він має роз'єм для підключення зовнішніх

пристроїв, як правило, роз'єм для зв'язку з комп'ютером та для програмування пам'яті мікроконтролера. Однією з особливостей мікроконтролера Atmel ATmega є можливість програмувати його без використання спеціального програміста. Тому все, що потрібно для створення нового електронного пристрою - це плата Arduino, кабель зв'язку та комп'ютер. Розробники можуть використовувати готові плати розширення або підключати необхідні компоненти безпосередньо до Arduino. У цьому випадку вся інша робота буде спрямована на розробку та коригування програм управління мовами програмування високого рівня. Можливість використання Arduino обмежена лише функціональністю мікроконтролера та наявними реалізаціями платформи.

Серія плат розвитку Arduino має більше десяти реалізацій, і їх технічні характеристики різні. Для кожного проекту можна вибрати найбільш підходящий варіант, і його функція відповідатиме задачі і буде використана в найбільшій мірі. Зокрема, серія апаратних платформ Arduino пропонує такі моделі:

- Платформа, що базується на 32-розрядному мікропроцесорі Cortex-M3 ARM SAM3U4E;
- Платформа на мікроконтролері Leonardo-ATmega32U4;
- Uno-платформа на мікроконтролері ATmega328, на даний момент одна з найпопулярніших платформ;
- платформа мікроконтролера Diecimila-ATmega168;
- Nano - невелика платформа на мікроконтролері ATmega168 або ATmega328;
- Платформа на мікроконтролері Mega2560-ATmega2560, що використовує чіп ATmega8U2 для послідовного зв'язку через порт USB;
- Версія платформи Mega ADK-Mega 2560, яка підтримує хост-інтерфейс USB для підключення телефонів на базі Android та інших пристроїв через інтерфейс USB.
- Платформа на мікроконтролері Mega-ATmega1280;

- Arduino Bluetooth - це платформа на мікроконтролері ATmega328. Його функція - наявність на платі модуля Bluetooth BT-V06, який може використовуватися для бездротового зв'язку та програмування пам'яті мікроконтролера.
- LilyPad - платформа на мікроконтролері ATmega168V або ATmega328V, спеціально розроблена для перенесення та вшивання в тканину.
- Платформа на мікроконтролері Fio-ATmega328 для бездротових додатків. У комплект Fio входить роз'єм для підключення радіомодулів XBee, роз'єм для акумуляторів LiPo та вбудована схема зарядки.
- Платформу Pro на мікроконтролерах ATmega168 або ATmega328 можна використовувати як частину великих проектів.
- Платформа Pro Mini-Microcontroller ATmega168 або ATmega328 та платформа Pro, можуть використовуватися як частина великих проектів, ці проекти вимагають низької ціни, невеликих розмірів та високої функціональності [7].

Оскільки цей проект не використовує інтерфейси Bluetooth, Ethernet, Wi-Fi, USB для взаємодії з іншими пристроями, рекомендується розглянути лише найпростішу та найменшу модель Arduino та вибрати найкращий варіант для поточної розробки.

2.4.2. Мікроконтролери PIC

Мікроконтролери PIC з Гарвардською архітектурою, розроблені компанією Microchip Technology Inc з США. Назва PIC є аббревіатурою від Peripheral Interface Controller, що означає «контролер периферійного інтерфейсу». У назві зазначено, що PIC спочатку був розроблений для розширення можливостей 16-розрядного мікропроцесора CP1600 [19]

В каталозі Microchip Technology Inc представлені різні 8-бітові, 16-бітові та 32-бітові мікроконтролери та цифрові контролери сигналу марки. Унікальною особливістю контролера PIC є хороша сумісність різних серій. Це

сумісність програмного забезпечення (єдине вільне середовище для розробки MPLAB IDE, компілятор GCC) та сумісність з точки зору контактів, периферійних пристроїв та напруг живлення при розробці найпопулярніших засобів протоколів зв'язку, бібліотек та стеків. Асортимент включає більше 500 різних контролерів, а також усі можливі варіанти периферійних пристроїв, пам'яті, кількості контактів, вихідної потужності, діапазону потужності та температури.

Перше, на що звернули увагу при розробці контролерів PIC, це простота та ефективність. Концепція PIC є загальною для всіх розроблених серій, заснованих на архітектурі RISC (комп'ютер із зменшеним набором команд), з простою командною системою, вбудованою пам'яттю для програм та даних, низьким енергоспоживанням.

Архітектура RISC базується на наступних основних принципах:

- будь-яка операція здійснюється за такт;
- система команд має містити мінімальну кількість команд однакової довжини;
- операції по обробці даних здійснюються лише у режимі "регістр-регістр";
- результат має створюватись з розрахунку одне слово на такт

Система команд базової серії PIC165х містить 33 команди. Це також зіграло свою роль у популярності контролерів PIC. Всі команди (за допомогою команд переходу) виконуються за один машинний цикл (або чотири машинні такти), здійснюючи перекриття часу вибору та виконання команд, що дає змогу досягти продуктивності до 5 MIPS при частоті тактів 20 МГц. [20]

Мікроконтролер PIC має симетричну систему команд. Це забезпечує можливість виконувати операції з довільним регістром і використовувати довільний метод адресації. Однак розробники Microchip не можуть відмовитись від структури, яка подобається всім через регістр-акумулятор, він є необхідним учасником усіх операцій із використанням двох операндів.

Користувач тепер може вибрати для збереження результатів операцій як регістр-акумулятор, так і робочий регістр.

Більшість контролерів PIC можуть мати одноразову програмовану програмну пам'ять за допомогою програмного забезпечення або шаблону ПЗП. Для налагодження пропонується версія із стиранням за допомогою ультрафіолетових випромінювань, але її ціна недешева. Загальна кількість опублікованих модифікацій контролера PIC становить приблизно 500 найменувань. Продукція компанії охоплює весь спектр використання 8-розрядних мікроконтролерів. [20]

Особливий акцент Microchip робить на максимальному зменшенні потужності виробленого мікроконтролера. Мікроконтролер має підтримку режиму сну, діапазон напруги живлення контролера PIC становить 2,0 ... 6,0 В.

Основна архітектура складається з контролерів серії PIC10 та деяких контролерів серії PIC12 та PIC16. Заснований на 12-бітній архітектурі програмного забезпечення, представлені контролери в корпусах від 6 до 28 контактів. Спрощена архітектура базової серії забезпечує найдешевше рішення, запропоноване Microchip. Характеризуються широким діапазоном напруг джерела живлення, здатні працювати при низькій напрузі, призначені для використання мікроконтролерів в акумуляторних пристроях.

Переваги:

- малі корпуси з малою кількістю виводів
- пам'ять програм, яку можна перезаписувати
- низька ціна
- легкий в освоєнні, має 35 команд
- ефективно здійснюють прості математичні операції;
- мають розвинену периферію;
- споживають мало енергії.

У широкому спектрі продуктів можна використовувати мікроконтролери в пристроях, розроблених для різних застосувань. На даний

момент Microchip випустив п'ять базових серій 8-бітних мікроконтролерів RISC із сумісним програмним кодом знизу вгору [19]:

- PIC12CXXX - серія мініатюрних 8-контактних версій мікроконтролерів. Ці мікроконтролери можуть бути використані в 12-розрядних та 14-розрядних командних системах. Містить вбудований годинник, таймер/лічильник, сторожовий таймер, управління перериваннями. Ця серія включає мікроконтролери з вбудованими 8-бітними АЦП з чотирма каналами. Може працювати при напрузі 2,5 В;
- PIC16C5X - базова серія мікроконтролерів з командами розміром 12-біт, виготовлені реалізації у 18-, 20- та 28-контактних корпусах. Це простий і недорогий мікроконтролер з мінімальним периферійним обладнанням. Робота при малих напругах живлення (до 2 В) дозволяє використовувати мікроконтролери в портативних приладах. Крім того дана серіє включає групу мікроконтролерів PIC16HV5XX, які можуть бути використані акумуляторами з напругою до 15 В;
- мікроконтролери PIC16CXXX з командами розміром 14-біт. Дана серія оптимізована для поєднання з різною периферією, включаючи аналогові компаратори, аналого-цифрові перетворювачі, контролери послідовного інтерфейсу SPI, USART та I2C, лічильники, модулі захоплення/порівняння та модулі супервізорів;
- високопродуктивні мікроконтролери PIC17CXXX розширеними системними командами розміром 14-біт. Вони можуть працювати на частоті до 33 МГц. На додаток до широкого діапазону периферійних пристроїв, 16-рівневого апаратного стеку та системи векторних переривань, більшість мікроконтролерів даної серії містять апаратний множник 8x8, що дозволяє виконувати операцію множення за один машинний цикл. Це один з найшвидших мікроконтролерів на 8 розрядів у відповідній ринковій ніші;
- PIC18CXXX - це серія мікроконтролерів з високою продуктивністю, мають розширену систему інструкцій розміром 16-біт та вбудованим 10-

бітним АЦП з робочою частотою до 40 мегагерц. Вони містять апаратний стек в 31 шар, до 4 КБ пам'яті даних та до 2 МБ можливостей адресації програм у зовнішній пам'яті. Ядро RISC в цій серії розширене і оптимізоване для нового компілятора мови C. [19].

Найпоширенішими контролерами серії PIC є PIC16CXXX та PIC17CXXX. Для програмування мікроконтролерів серії PIC використовуються програмні засоби IC PROG, ICD-2, ICD-3, REAL ICE, Pickit, PicKit2, Pickit3. Ці програми дозволяють програмувати та налагоджувати код: поетапно виконувати, дозволяють робити точки зупинки, дозволяють бачити вміст оперативної пам'яті та пам'яті програм, стеку.

2.4.3. Мікроконтролери STM32

STM32 – мікроконтролер, в основу якого покладено ядро ARM Cortex-M3 [21]. Воно має велику кількість позитивних сторін, але на сьогоднішній день його основна перевага це універсальність. За два роки ядро Cortex-M3 стало стандартом для відповідної сфери. Кількість виробників, які перешли на його використання підтверджують це. Більшість виробників мікроконтролерів, за винятком Microchip, розробляли раніше або розробляють рішення на основі даної архітектури, зокрема Texas Instruments, STMicroelectronics, ATMEL, NXP, Renesas Electronics тощо. STMicroelectronics була серед перших компаній, яка випустила мікроконтролер на основі Cortex-M3 та швидко завоювала велику частку ринку.

Серія STM32 була випущена в 2010 році. До цього STMicroelectronics випустила 4 серії мікроконтролерів на базі ARM, але їх продуктивність гірша. Факти довели, що контролери STM32 є найкращими з точки зору продуктивності та ціни. Спочатку вони були доступні в 14 версіях, розділених на 2 групи – з тактовою частотою до 2 МГц, та до 36 МГц. Програмне забезпечення та контакти в двох групах мають аналогічні. Початковий продукт

має 128 КБ вбудованої флеш-пам'яті та 20 КБ оперативної пам'яті. Зараз лінійка продуктів значно розширена, і з'явилися нові представники зі збільшеними значеннями оперативної та пам'яті програм. [22]

Переваги STM32

- низька вартість;
- багато середовищ для розробки;
- мікросхеми взаємозамінні - якщо у вас недостатньо ресурсів для мікроконтролера, можна замінити його на більш потужний мікроконтролер, не змінюючи схему та друковану плату.
- висока продуктивність;
- зручні налаштування мікроконтролера.

Недоліки STM32:

- високий бар'єр для входу;
- наразі літератури про STM32 мало.
- більшість створених бібліотек застаріли, що зумовлює необхідність створення власних.

Серія STM32 має широкий асортимент товарів, які відрізняються пам'яттю, продуктивністю, енергоспоживанням та іншими характеристиками. STM32F-1, STM32F-2 та STM32L серії повністю сумісні. Кожна серія має десятки чіпів, і їх можна легко замінити на інші продукти. STM32F-1 - це перша лінійка, і їх продуктивність обмежена. Тому контролера швидко наздогнали серії Stellaris та LPC17 з точки зору продуктивності. Пізніше випущена вдосконалена STM32F-2 - тактова частота досягала 120 МГц. Вона має високу обробну потужність завдяки технології виробництва 90 нм. Продукція серії STM32L представлена моделями, виготовленими за спеціальним процесом, втрати транзисторів дуже малі, тому мікроконтролер демонструє кращу вартість. [23]

Контролер STM32W не сумісний з контактами контролерів STM32F-1, STM32F-2 та STM32L. Причина в тому, що лінійка була розроблена компанією, яка надає радіочастотну частину. Це накладає обмеження на розвиток

мікроконтролерів. Мікросхема STM32F100R4 має малий набір функцій. Об'єм флеш-пам'яті становить 16 КБ, оперативної пам'яті - 4 КБ, тактова частота - 12 МГц. Якщо потрібен швидший пристрій, який має збільшену до 128 КБ флеш-пам'ять, піжійде STM32F101RB. Продукт STM32F103RE реалізує в собі інтерфейс USB. [24]

2.5. Висновок розділу

На основі ключових модулів пристрою, описаних у першій частині, викладено можливу реалізацію цих модулів. Зокрема, розглядаються різні типи плат Arduino та їх функції. Розглянуті певні типи датчиків температури та вологості з урахуванням різних випарювачів води.

3. Технології та компоненти, що використовуються для реалізації пристрою

3.1. Вибір мікроконтролера для приладу

Були розглянуті мікроконтролери трьох виробників: Atmel, Microchip та STMicroelectronics. Кожен з них має особливості реалізації та використання, також для кожного на ринку представлена плата для розробки, яка полегшує розробку приладів на основі відповідного мікроконтролера.

Можливості мікроконтролерів виробництва Microchip та STMicroelectronics більші, ніж виробництва Atmel. По співвідношенню ціни і характеристик найкращі мікросхеми Atmel. Крім того в мікроконтролерів Atmel нижчий поріг входу порівняно з виробами Microchip та STMicroelectronics. Це пов'язано з поширеністю платформи Arduino для розробки. Тому виберемо мікроконтролери компанії Atmel, а для розробки використаємо Arduino.

Бездротовий інтерфейс не використовується для поточної розробки, не потребує зв'язку з телефонами та іншими периферійними пристроями USB, не вимагає автономності та високої швидкості, тому підходить найпростіша реалізація Arduino.



Рисунок. 3.1 – Arduino Nano

Для розробки ми оберемо платформу Arduino Nano, оскільки вона відрізняється від найпростішої платформи Arduino тим, що на материнській платі є схема програмування пам'яті мікроконтролера і можливість підключення до персонального комп'ютера через USB. 14 цифрових входів/виходів платформи достатньо для зв'язку та управління всіма електронними компонентами та пристроями, які будуть підключені до друкованої плати. Плата розробки також може бути підключена до зовнішнього джерела живлення до 20 вольт, таким чином уникаючи необхідності розробляти додатковий перетворювач напруги з виходом 5 В для Arduino. Це важливо, коли схема містить пристрої з напругою живлення вище 5В (наприклад, вентилятори, лампи розжарювання, світлодіодні стрічки тощо).

Платформа Arduino Nano дуже мала і може бути використана для простих проектів. Платформа базується на мікроконтролерах ATmega328 (Arduino Nano 3.0) або ATmega168 (Arduino Nano 2.x).

Arduino Nano може житися від USB Mini-B, зовнішнього джерела живлення напругою від 7 до 20 В (через контакт 30) або джерела живлення зі стабільною напругою 5 В (через контакт 27). Коли платформа підключена до декількох джерел живлення, вона буде житися від джерела живлення з найвищою напругою.

У таблиці наведено характеристики платформи Arduino Nano. 2.3 [8].

Таблиця 3.1 – Особливості Arduino Nano

Мікроконтролер	ATmega168 або ATmega328
Робоча напруга	5 В
Рекомендована вхідна напруга	7-12 В
Максимальна вхідна напруга	6-20 В
Цифрові входи/виходи	14 (6 з яких можна використовувати як вихід ШІМ)
Аналогові входи	8

Продовження таблиці 3.1 – Особливості Arduino Nano

Максимальний постійний струм вхідів/виходів	40 мА
Максимальний постійний струм виводу 3,3 В	50 мА
Розмір пам'яті	16 КБ (ATmega168) або 32 КБ (ATmega328), з яких 2 КБ використовується завантажувачем
Обсяг оперативної пам'яті	1 КБ (ATmega168) або 2 КБ (ATmega328)
Розмір EEPROM	512 байт (ATmega168) або 1024 байта (ATmega328)
Тактова частота	16 МГц

3.2. Вибір екрану для виведення інформації

Для відображення інформації можна використовувати семисегментний індикатор та рідкокристалічний дисплей (далі - РК). Розглянемо переваги та недоліки кожного виду.

До переваг семисегментних індикаторів можна віднести відносно просте управління, можливість читати їх при яскравому світлі та можливість відображати всі цифри та більшість букв. До недоліків можна віднести те, що вони громіздкі, взагалі не можуть відображати інформацію без випромінювання світла і потребують зовнішнього декодера.

До переваг РК-дисплеїв можна віднести можливість відображення будь-яких символів, можливість роботи без підсвічування, компактність та інтерактивність з ними, адже для передачі на дисплей достатньо кодів символів та координат, а вбудований контролер визначатиме пікселі, які потрібно засвітити.

Зрозуміло, що використовувати послідовний порт в якості монітора виведення даних не зовсім зручно, а для автономних проектів неможливо. Розумно для цієї мети, а також для виведення інформації при налагодженні створюваних проектів застосувати рідкокристалічний індикатор. Рідкокристалічні індикатори (PKI) різних фірм (POWERTIP, MICROTIPS, WINSTAR, Fordata, Anshan Yes, Data Vision, Newtec Sunlike) впевнено завоюють світовий ринок електроніки сьогоднішнього дня і поступово витісняють старі добрі стрілочні прилади. Сучасному розробнику апаратури для фізичних експериментів і промисловості необхідно знати основні принципи роботи цих зручних і корисних пристроїв. Всі рідкокристалічні індикатори, що випускаються в різних країнах, з точки зору програміста однотипні. Але інструкції, що даються розробниками, зазвичай написані англійською мовою, скупі і важкі для розуміння. Проте слід врахувати, що в пристроях, що випускаються в різних країнах, спостерігається явна тенденція до стандартизації, і практично всі випущені ЖК-індикатори орієнтовані на застосування контролера HD44780.

Підберемо недорогий, але в той же час надійний, символічний індикатор досить великого розміру для виведення великого обсягу інформації. У проектах Arduino часто використовують недорогі індикатори фірми WINSTAR Display Co., Ltd. - одного з найбільших світових розробників і виробників високоякісних символічних і графічних (в т. ч. кольорових) ЖК-індикаторів. Штаб-квартира компанії знаходиться на Тайвані, а виробничі потужності, розташовані на площах понад 3000 кв. метрів, забезпечують випуск понад мільйон різних індикаторів в рік. Виробництво сертифіковане по системі ISO 9001.

Для цього пристрою вибираємо РК-дисплей, що відображає два рядки символів, кожен рядок може відображати 16 символів. Завдання він повністю задовольняє. Для відображення пунктів меню та параметрів відображення достатньо 32 символів. Цей вибір підходить, оскільки семисегментний індикатор не дозволяє відображати спеціальні символи, такі як піктограми

градуси та відсотки. З іншого боку, пристрій не відображає складної графічної інформації (графіки, графіки тощо), тому недоцільно вибирати РК-матрицю з високою роздільною здатністю.

У поточному обладнанні ми будемо використовувати символний РК-модуль RC1602B виробництва Raustar. Він працює на контролері ST7066 [26] і може бути замінений дисплеєм з тим самим або подібним контролером.

3.3. Вибір датчика температури для виконання завдання

Піроелектричні датчики не підходять для поточної задачі, оскільки їх діапазон вимірювання набагато більший, ніж діапазон, який вимагає проєктований прилад. Це призведе до неефективного використання, а ціна датчиків термопар порівняно висока. Кварцові датчики використовуються для відносного вимірювання разом з іншими датчиками і мають більш високу температурну роздільну здатність, яка набагато перевищує температурну роздільну здатність, необхідну для виконання завдання, що призводить до неефективного використання його функцій і, отже, неефективного використання грошових ресурсів. Датчик теплового опору повністю відповідає вимогам вимірювання.

Очевидно, напівпровідники кращі за метали в необхідному діапазоні, оскільки їх діапазон температур менший, ніж у металів, і вони можуть повністю відповідати вимогам. Але більшість напівпровідникових датчиків пропонуються у формі мікросхем без спеціального захисту від їх агресивного середовища, тому слід вибирати датчики температури з металу, оскільки вони завжди укладені в металевий кожух, такий як капсула з нержавіючої сталі.

Датчик температури PT100 вибраний з доступних на ринку металевих терморезистивних датчиків. Він виготовлений з платини і має опір 100 Ом при температурі 0 °C. Виміряна температура коливається від -20 °C до + 500 °C, з допустимим відхиленням до 0,3 °C і допуском донизу 0,8 °C. Коефіцієнт опору терморезистора дорівнює 0,39 Ом/°C, тобто з кожним градусом поточний опір

буде збільшуватись на 0,39 Ом, що свідчитиме про відповідне підвищення температури. Датчик циліндричний, діаметром 4 мм і висотою 30 мм, а корпус виготовлений із сталі [27].

3.4. Вибір випарника

Обраний випарник повинен відповідати вимогам щодо ефективного зволоження і одночасно може використовуватися як інгалятор. Хоча випарник гарячої пари може ефективно зволожувати, він має недолік - побічним ефектом є нагрівання повітря, і якщо не дотримуватись умов експлуатації, існує ризик опіків. Як інгалятор його не можна використовувати, оскільки температура пари на виході висока, що неприпустимо під час вдихання. Тому необхідно вибирати серед випарників холодної пари. Компресорний випарник можна використовувати лише як випарник інгалятора, оскільки він має достатню продуктивність і може використовувати будь-які ліки. З іншого боку, він категорично не підходить для зволоження повітря, оскільки створює високий рівень шуму, що шкідливо при тривалому впливі на людину. Ультразвуковий випарник найбільш повно відповідає всім вимогам. Він має нижчий рівень шуму, який викликаний лише взаємодією дрібних частинок води. Його продуктивності достатньо для обслуговування менших приміщень, одночасно можна використовувати як інгалятор. Кінцеве використання базується на тому, що розмір частинок розпилювача становить 2-3 мкм, що дозволяє цим частинкам дістатися до нижніх дихальних шляхів і доставити туди препарат. Ультразвуковий генератор туману, що використовується в цій роботі, має напругу живлення 24 В, 20 Вт і вбудований датчик рівня води.

3.5. Вибір датчика вологості

Розглянемо кожен тип датчика вологості в рамках цього питання.

Ємнісний датчик вологості підходить для поточного завдання. Він має багато переваг, включаючи низьку вартість, лінійні характеристики та невеликі розміри. Крім того, діапазон вимірювань дуже широкий, охоплює можливі значення вологості в приміщенні. Недоліком є необхідність генератора частоти, який визначає точність вимірювань.

Резистивні датчики мають високу точність та невеликі розміри, тому вони можуть бути використані в даному приладі. Ці датчики мають досить широкі діапазони вимірювань, тому вони цілком можуть бути використані для вимірювання вологості в приміщеннях. У проектованому приладі опір датчика може бути визначений безпосередньо контролером з використанням невеликої обв'язки і далі його можна перетворити у показники відносної вологості.

Отже, як конденсатори, так і датчики опору можуть відповідати вимогам конструкційного обладнання. Рекомендується використовувати комбінований датчик вологості, який містить датчик вологості та схему перетворення сигналу. Датчик DHT22 обраний з широко доступних. Це комбінований датчик вологості, що включає датчик вологості, схему, яка безпосередньо перетворює вихідний сигнал у значення вологості, і датчик температури, який вимірюватиме температуру в приміщенні. Виміряне значення від датчика передається по одному проводу [28]. Його можна прочитати з цифрового входу Arduino.

3.6. Вибір вентилятора

Ультразвукові парогенератори можуть лише перетворювати воду в пару, але за відсутності зовнішніх можливостей для генерації пари вона осяде назад у воду. Вентилятор повинен бути вентилятором без ШІМ і встановлюватися збоку від схеми управління, оскільки він може працювати в умовах підвищеної

вологості та контактувати з парою, а нормальна вологість для електронних компонентів переважно становить 40-60%. Рекомендується використовувати вентилятор постійного струму з напругою живлення 24 В. Максимальний розмір - 40x40 мм, оскільки найменший потік повітря достатній для виштовхування виробленої пари. Однак насправді важко знайти вентилятор з таким малим розміром і напругою живлення 24 В, тому вибираємо вентилятор з напругою 12 В. Обраний вентилятор Sunon EB40101S2-000U-999. Він має напругу живлення 12 В і потужність 1,08 Вт, що може повністю вирішити проблему [29].

3.7. ШІМ

3.7.1. ШІМ-регулювання потужністю

ШІМ може бути використаний для управління величиною потужності, що подається на навантаження без втрат за рахунок контролю подачі енергії через опір. Недоліком цієї технології є те, що потужність, що споживається навантаженням, не є постійною, а переривчастою, а енергія, що подається на навантаження, не є постійною. Однак навантаження може бути індуктивним і мати досить високу частоту, тому при необхідності інші пасивні електронні фільтри можуть бути використані для згладжування імпульсу та отримання аналогової форми сигналу. Потужність, що надходить у навантаження, може бути постійною. Струм від джерела живлення переривчастий, і в більшості випадків енергію потрібно буде накопичувати на стороні джерела живлення.

Використання напівпровідникових перемикачів дозволяє легко реалізувати ШІМ-систему височастотного управління потужністю. Як зазначалося вище, мінімальна потужність розсіюється перемикачем у стані увімкнення або вимкнення. Однак під час переходу між увімкненим та вимкненим станами напруга та струм не дорівнюють нулю, тому потужність розсіюється в перемикачі. У разі швидких змін стану між повністю відкритим і закритим (як правило, менше 100 наносекунд) споживання енергії в

комутаторі може бути дуже низьким порівняно з потужністю, що подається на навантаження [25].

Сучасні напівпровідникові перемикачі, такі як транзистори МОН або біполярні транзистори із ізольованими затворами, дуже підходять для високопродуктивних контролерів. Ефективність перетворювача частоти, що використовується для управління двигуном змінного струму, може досягати більше 98%.

Контролери швидкості вентилятора, що регульованою швидкістю, зазвичай використовують ШІМ, оскільки він набагато ефективніший, ніж змінні резистори. До того ж останній навряд чи може працювати в електронній формі, для цього потрібен невеликий двигун.

Ці досить прості типи димерів можна ефективно використовувати з інертними (або відносно повільними) джерелами світла, такими як лампи розжарювання. Регулювання потужності може спричинити лише незначні додаткові коливання оптичного випромінювання. Деякі інші типи джерел світла (наприклад, світлодіоди) вмикаються і вимикаються дуже швидко, і будуть мерехтіти при живленні на низьких частотах. Збільшуючи частоту ШІМ, можливі ефекти мерехтіння від таких швидко реагуючих джерел світла можуть бути зменшені. Якщо світло коливається досить швидко, зорова система людини не зможе їх сприймати, а очі сприйматимуть середню інтенсивність без мерехтіння [25].

3.7.1. Робота з ШІМ в Arduino

В платформах Arduino присутній апаратний ШІМ. Для його використання потрібно підключити навантаження до відповідного контакту з підтримкою технології так налаштувати програмно роботу ШІМ на цьому контакті.

Arduino Nano побудована на основі ATmega328. Вона реалізує ШІМ з 8-бітної роздільної здатності. Підтримка ШІМ реалізована на цифрових

контактах 11, 10, 9, 6, 5, 3. Таким чином на виході портів доступні 256 значень напруги, які задовольняють більшість задач. При живленні 5 вольт напругу на виході можна змінювати з кроком $5\text{В}/256 \approx 0,0195\text{ В} \approx 19,5\text{ мВ}$. В середовищі розробки Arduino доступні функція `analogWrite()`, яка дозволяє задати величину напруги на потрібному порті:

```
analogWrite(pin, value);
```

"pin" - номер виходу, на який потрібно отримати сигнал ШІМ; "value" - значення від 0 до 255. Це значення визначає середнє значення напруги періоду ШІМ-сигналу [8].

3.8. Висновок розділу

У цьому розділі представлена технологія ШІМ для регулювання потужності навантаження та її використання в Arduino. У розробленому обладнанні вона буде використовуватися для регулювання потужності ультразвукового випарника. Відповідно до параметрів обладнання вибирається плата розробки Arduino, яка відповідає мінімальним вимогам. Вибраний екран відображає інформацію про параметри та режими роботи. Це дворядковий символьний дисплей із 16 стовпцями на рядок. Відповідно до вимог, вибраний водонепроникний датчик температури для вимірювання температури води, що вливається в пристрій. Також було обрано ультразвуковий пристрій для випаровування води, резистивний датчик вологості з ланцюгом перетворення та вентилятор для відводу утвореної пари.

4. Схема роботи приладу та керуючий код

4.1. Програмування в Arduino

4.1.1. Цифрові входи

Контакти платформи Arduino можна використовувати як вхід або вихід. Аналоговий вхід Arduino (ATmega) також можна налаштувати і експлуатувати так само, як цифровий порт вводу-виводу.

Контакти Arduino налаштовані як вхідні порти, тому немає необхідності оголошувати їх у функції `pinMode()`. Налаштований вхідний порт знаходиться у стані високого імпедансу. Це означає, що вхідний порт надає невелике навантаження на схему, яка його містить. Менше значення потрібно для перемикання вхідного порту з одного стану в інший. Якщо вихідного з'єднання немає, значення на ньому буде отримано випадковою величиною, спричиненою електричним шумом.

Якщо сигнал на вхідний порт не надходить, рекомендується встановити порт у відомий стан. Цього можна досягти, додавши 10 кОм підтягуючий резистор, який включає в себе джерело живлення +5 В або заземлення. Мікроконтролер ATmega має програмований вбудований підтягуючий резистор 20кОм. Програмування цих резисторів є таким:

```
pinMode(pin, INPUT); // призначити порт входом  
digitalWrite(pin, HIGH); // підключити підтягуючий резистор
```

Контакт, налаштований як вихідний порт, знаходиться в стані низького імпедансу. Через ці контакти протікає незначний струм. Контакти мікросхеми ATmega можуть бути джерелами струму до 40 мА. Для більшості реле, соленоїдів та двигунів цього поточного значення недостатньо.

Коротке замикання пінів Arduino або спроба підключення енергоємних пристроїв може пошкодити вихідні транзистори або весь мікроконтролер ATmega.

4.1.2. Аналогові входи

Мікроконтролер ATmega, який використовується в Arduino, включає шестиканальний аналого-цифровий перетворювач (АЦП). Роздільна здатність перетворювача становить 10 біт, що дозволяє на виході отримувати значення від 0 до 1023. Аналогові входи можна використовувати як цифрові контакти для портів вводу-виводу, і їх кількість становить від 14 до 19:

```
pinMode(14,OUTPUT);  
digitalWrite(14, HIGH);
```

Для виводу, який раніше використовувався як цифровий вихідний порт, команда `analogRead` не працюватиме належним чином. У цьому випадку рекомендується налаштувати його як аналоговий вхід.

4.1.3. Функції `setup()` і `loop()`

Основна структура програми Arduino містить щонайменше дві основні частини: функцію `setup()` та `loop()`. Перед функцією `setup()` здійснюється оголошення змінної, підключення бібліотек. Кожного разу, коли плата Arduino включається або скидається, функція `setup()` запускається один раз. Вона використовується для ініціалізації змінних, встановлення режимів порту та інших операцій підготовки основного робочого циклу. Навіть якщо вона нічого не робить, її потрібно включити в програму.

Функція `loop()` у нескінченному циклі послідовно виконує команди, описані в її тілі. Ця функція виконується циклічно, і вона виконує основну роботу.

4.2. Робота з датчиком вологості

Як згадувалося вище, передача даних датчика DHT22 здійснюється по одній лінії. Вихідний порт - це другий контакт датчика. Він повинен бути

підключений до будь-якого цифрового або аналогового цифрового входу Arduino. Підключаємо його до дев'ятого цифрового контакту. Для взаємодії з датчиком, в тому числі для конфігурування контакту зв'язку, ініціалізації та зчитування інформації з нього, повинна бути підключена бібліотека "DHT.h". Вона призначений для взаємодії з датчиками модельного ряду DHT [8].

Наступним кроком підключення датчика є встановлення відповідного контакту Arduino у режим входу та ініціалізація, тобто подача сигналу запуску датчика. Ці кроки виконуються в коді в розділі `setup()`:

```
pinMode(DHT_PIN, INPUT);
dht.begin();
```

Отже, функція `pinMode()` дозволяє призначити будь-який контакт для введення або виведення, а `dht.begin()` запустить датчик вологості. Далі потрібно зчитувати температуру і вологість з датчика. Ми зробимо це відразу після першої ініціалізації в розділі `setup()`:

```
currentHumidity = dht.readHumidity();
t = dht.readTemperature();
```

Наступні читання будуть виконуватися в циклі в розділі `loop()`, який буде виконувати наступний код:

```
if( (( millis() - delayOfRead) > 1000 ) ) {
    currentHumidity = dht.readHumidity();
    t = dht.readTemperature();
    delayOfRead = millis();
}
```

Тут, крім безпосереднього зчитування параметрів вологості та температури, зроблена затримка зчитування тривалістю 1000 мс. Це пояснюється тим, що датчик DHT22 можна перевіряти не частіше одного разу на секунду. Слід зазначити, що температура та вологість зчитуються з точністю до двох знаків після коми, але для відображення достатньо зчитати температуру з точністю одного знаку після коми, значення вологості достатньо взяти без дробової частини.

4.3. Робота з датчиком температури

Термістор PT100 не може бути безпосередньо підключений до Arduino через два контакти, оскільки ланцюг не буде працювати, оскільки на вході платформи немає амперметра, тому законом Ома не зможемо визначити опір датчика. Необхідно визначити напругу між резисторами на основі схеми дільника напруги в парі з постійним резистором та використовувати аналого-цифровий перетворювач (далі - АЦП). Постійний опір вибираємо найближчим до опору датчика. Більш високий опір зменшить точність вимірювання опору, оскільки чим вищий опір порівняно зі зміною температури, тим менша зміна опору датчика. Arduino АЦП може не розпізнавати незначні зміни напруги. Крім того, резистор з постійним опором із більшим опором має великі варіації номінального значення опору, навіть з невеликим відхиленням у відсотках. Отже, резистор з опором 1000 Ом і допуском 1% матиме відхилення опору 10 Ом, що є достатнім для спотворення вимірюваної напруги. Якщо вибрати резистор з меншим опором, бо це зменшить падіння напруги на ньому, а це збільшить величину струму через датчик та резистор. Цей струм може спричинити нагрівання датчика і, відповідно, спотворення результатів вимірювань. Виходячи з вищесказаного, постійний резистор R3 в дільнику напруги вибирається рівним 100 Ом з допуском 1%. Вибір допуску менше 1% безглуздий, оскільки такі резистори дорогі і не підходять для цього завдання. Також немає необхідності вибирати резистор з допуском більше одного відсотка, оскільки він сильно вплинув би на результати.

Тільки тоді, коли опір постійного опору вимірюється прецизійним вимірювальним приладом, і отримане значення відповідає формулі для розрахунку опору, можна отримати найкращу точність вимірювання опору за допомогою дільника напруги.

В результаті ми отримуємо схему підключення термістора, як показано на малюнку 4.1. У нашому випадку вимірювання напруги в середній точці здійснюється через аналоговий вхід Arduino A0. Для цього можна

використовувати аналоговий контакт, але не можна використовувати цифровий вхід, бо він вміє зчитувати лише високий та низький логічні рівні.

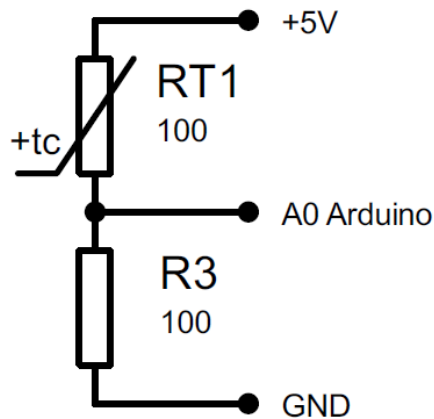


Рисунок 4.1 – Підключення термістора

Потужність, яку споживатиме датчик температури розрахуємо відповідно до закону Ома. Струм через дільник напруги:

$$I = \frac{U}{RT1 + R3} = \frac{5}{100 + 100} = 0,025 \text{ (A)}; \quad (4.1)$$

Підставляємо отримане значення у формулу:

$$P = I^2 R = (0,025)^2 \cdot 100 = 62,5 \text{ (мВт)}. \quad (4.2)$$

Враховуючи, що розмір датчика великий, а його корпус виготовлений із сталі з високою теплоємністю, цієї потужності недостатньо для значного спотворення результатів вимірювань. Тому обране рішення можна вважати вдалим. Формула знаходження напруги на термісторі для даного дільника напруги:

$$U_{RT1} = U_{оп} \frac{R_{RT1}}{R_{RT1} + R_3}; \quad (4.3)$$

де U_{RT1} – напруга на терморезисторі; $U_{оп}$ – опорна напруга; R_{RT1} і R_3 – значення опорів. Опір R_{RT1} :

$$R_{RT1} = \frac{R_3}{\frac{U_{оп}}{U_{RT1}} - 1}; \quad (4.4)$$

Згідно з останньою формулою, ми знайдемо опір датчика температури, але спочатку нам потрібно визначити значення $U_{оп}$. У нашому прикладі опорна напруга дорівнює напрузі живлення Arduino, яка дорівнює 5 В, але АЦП Arduino Nano не може працювати у вольтах, а зчитує напругу у восьмирозрядному форматі. Отже, $U_{оп}$ слід приймати як 1023 - максимальне значення, яке можна прочитати АЦП [8].

Далі реалізуємо підключення датчика на рівні коду. У цьому випадку не потрібно підключати інші бібліотеки. Спочатку для зручності назовемо вхід A0 програмного коду з іменем WATER_TEMPERATURE_PIN, що відповідає його призначенню, і використаємо такий код на початку програми для роботи:

```
#define WATER_TEMPERATURE_PIN A0
```

Наступний код встановлює аналоговий контакт A0 у режим введення:

```
pinMode(WATER_TEMPERATURE_PIN, INPUT);
```

Далі потрібно прочитати напругу в середній точці і перетворити її на опір:

```
thermistorValue=resistorValue*(1023.0/analogRead(WATER_TEMPERATURE_PIN)-1);
```

thermistorValue - розрахунковий опір; resistorValue - значення опору на другому плечі дільника напруги; analogRead(pin) - прочитане значення аналогової напруги від 0 до 1023 на вході контакту.

Далі використаємо наступний код, щоб визначити температуру з опору:

```
currentWaterTemperature = (thermistorValue - defaultThermistorValue)/  
ohmByGradus;
```

де currentWaterTemperature – розрахункова температура;
thermistorValueDefault- опір термістора при 0°C; ohmByGradus - температурний коефіцієнт термістора.

Після завершення підключення термістора розрахована температура буде зчитуватися із змінної currentWaterTemperature.

4.4. Кнопки керування

Управління пристроєм здійснюватиметься за допомогою двох кнопок. Щоб уникнути додавання ще двох кнопок для підтвердження входу та виходу з режиму, ми будемо розрізняти коротке натискання та тривале натискання. Це дозволить отримати інший стан на кожній кнопці, загалом їх у нас чотири. Ці кнопки називаються "+" або "вгору" та "-" або "вниз". Якщо це меню інгаляції чи зволоження, достатньо коротко натиснути кнопку "+", щоб збільшити значення, або змінити пункти меню. Аналогічно, якщо це головне меню, коротко натисніть кнопку "-", щоб перемкнути пункт меню або зменшити значення у вибраному режимі. Натисніть і утримуйте кнопку "+", щоб перейти до обраного пункту меню або підтвердити зміни та перейти до наступного меню, залежно від того, яку частину меню натиснуто. Натисніть і утримуйте кнопку "-" у будь-якій частині меню, щоб повернутися до головного меню та зупинити випарник та вентилятор.

Була обрана тактова кнопка, оскільки в нашій схемі низькі струми і напруги. Крім того, цей тип кнопок дуже поширений. При проектуванні корпусу слід вибрати конкретну реалізацію наскрізного або поверхневого монтажу компонентів, але найкраще вибрати наскрізний монтаж, оскільки такі кнопки можна краще закріпити на платі і, отже, краще витримувати часті клацання.

Підключаємо кнопки до аналогових входів A4 та A5, що працюють у цифровому режимі. Коли на терміналі аналогового входу відсутній сигнал, то якщо ми зчитуємо значення з терміналу аналогового входу, то випадково отримаємо значення логічного нуля або одиниці. Тому, щоб вхід Arduino працював стабільно, необхідно підключити резистор 1-10 кОм. Цей резистор повинен іншим контактом забирати з входу паразитні сигнали на землю. Більший опір може не забезпечити потрібну стабільність, а менший буде негативно впливати на зчитуваний сигнал.

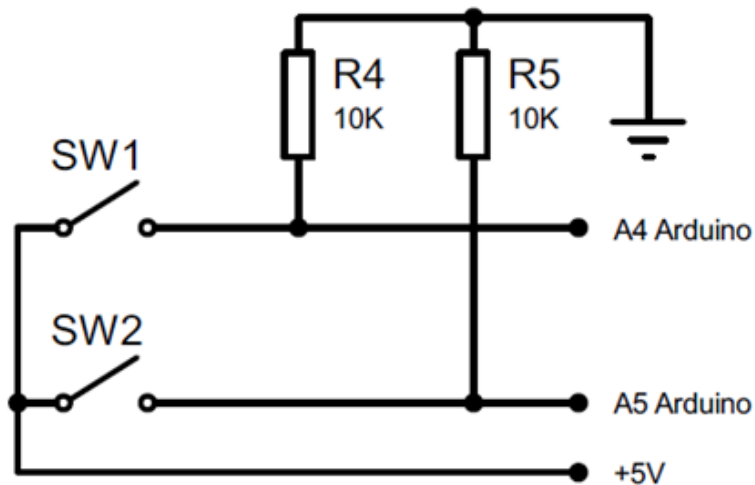


Рисунок 4.2 – Схема підключення кнопки та Arduino

Вибрано резистор з номінальним опором 10кОм. Цього достатньо для забезпечення чіткого логічного нульового рівня на вході. При натисканні кнопки через відповідний резистор буде протікати струм 0,5 мА. На основі вищезазначених даних було розроблено рішення для підключення кнопки до Arduino. Як показано на малюнку 4.2. Код для використання кнопки нижче:

```

if (digitalRead(BUTTON_UP_PIN) == HIGH && buttonUpFlag == 0) {
    UpTimeCounter = millis();    buttonUpFlag = 1;
    clickDelay = millis();
}
if ((millis() - clickDelay) > 50) {
    if (digitalRead(BUTTON_UP_PIN) == LOW && buttonUpFlag == 1) {
        unsigned long upTimeInterval = millis() - UpTimeCounter;
        buttonUpFlag = 0;
        if (upTimeInterval < timeInterval) {
            shortKeyPressedUp = true;
        } else {
            longKeyPressedUp = true;
        }
    }
}
}

```

Цей код обробляє натиски клавіші «+». В ньому відбувається зчитування двох станів кнопки: натиснутого і віджатого. Зчитування здійснюється із затримкою 50 мс, яка була підібрана експериментально. Затримка додана для уникнення випадкових натискань кнопки внаслідок нещільності контактів усередині кнопки. Також тут розрізняються довгі та короткі натиски клавіш шляхом оцінки тривалості натискань. Обробка роботи клавіші «-» аналогічна по структурі.

4.5. Використання дисплею

4.5.1. Фізичний зв'язок дисплею

Інформація відображатиметься на дисплеї RAYSTAR RC1602B. Опис контактів цього дисплея [26] наведено в таблиці 4.1.

З таблиці 4.1 видно, що дисплей живиться від перших двох контактів. Дисплей можна підключити до джерела напруги 3 В або 5 В. Підключімо його до того ж джерела живлення, що і Arduino. Ми також підключили 16 контактів до землі і підключили їх до Arduino через резистор 220 Ом через 15 контактів, щоб увімкнути підсвічування дисплея. На цьому малюнку він представлений як R6. Для регулювання контрастності використаний резистор зі змінним опором 10кОм. Підключімо дві крайні клеми до заземлення та джерела живлення 5 В відповідно, а середню точку підключімо до контакту Vo дисплея.

Таблиця 4.1 – Контакти RAYSTAR RC1602B

Номер контакту	Назва	Призначення
1	Vss	Мінус живлення
2	Vdd	Плюс живлення
3	Vo	Контраст
4	RS	Сигнал вибору даних/ команд

Продовження таблиці 4.1 – Контакти RAYSTAR RC1602B

5	RW	Сигнал вибору режиму читання/запису
6	E	Сигнал живлення
7	DB0	Шина даних
8	DB1	Шина даних
9	DB2	Шина даних
10	DB3	Шина даних
11	DB4	Шина даних
12	DB5	Шина даних
13	DB6	Шина даних
14	DB7	Шина даних
15	A	Плюс підсвітки
16	K	Мінус підсвітки

Після складання пристрою та ввімкнення підсвічування потрібно відрегулювати контрастність, оскільки при вимкненому підсвічуванні контраст може бути хорошим, а при увімкненні підсвічування контраст може значно зменшитися через яскравість пікселів. Критерієм гарної контрастності є те, що відображається текст добре видно, тоді як інші вимкнені пікселі невидимі. Вивід R/W повинен бути заземлений, оскільки в цьому пристрої ми не використовуємо функцію зчитування з дисплея, а лише записуємо на нього.

Дисплей може працювати в двох режимах: 8-бітний і 4-бітний. Для роботи в 8-розрядному режимі призначені старші та молодші біти DB0-DB7, а в 4-розрядному режимі призначені лише біти DB4-DB7. 8-бітний режим дозволяє оновлювати інформацію на екрані вдвічі частіше, ніж 4 біти, оскільки в останньому разі дані та інструкції передаються у два цикли: спочатку старші 4 біти, а потім молодші 4 біти. Цей дисплей може оновлювати дані до 10 разів на секунду. Тому в 4-розрядному режимі ми отримуємо близько 5 оновлень, яких достатньо для позначення режиму та параметрів, оскільки спроектований пристрій не повинен надавати індикацію значень, які швидко змінюються з

часом. Крім того, 4-бітний режим вимагає використання чотирьох контактів контролера для передачі даних та інструкцій. Контакти RS і E з'єднані з цифровими контактами [26]. Тому підключимо дисплей до Arduino наступним чином: RS до D7, E до D6, DB4-DB7 до D5-D2 та підсвічування A до D12.

Контролер РКІ може спільно з зовнішнім контролером виконувати різні дії: передавати дані по паралельному інтерфейсу в обох напрямках, змінювати напрямки заповнення знакомісць РК-дисплею, маніпулювати рядками і т. д. Після включення живлення робота РК-модуля завжди починається з процедури його ініціалізації, налаштування якої зберігаються до виключення живлення пристрою. До кінця процедури ініціалізації модуль РКІ знаходиться в стан готовності до роботи з даними. Якщо ініціалізація РК-модуля пройшла успішно, модуль готовий до роботи - для нього задано число рядків і розмір знакомісця (де знакомісце - це матриця 5×7 або 5×10 точок).

Таблиця 4.2 – Умови виконання операцій читання/запису

RS	R/W	E	Операція
0	0	1->0	Запис в регістр команд
1	0	1->0	Запис в регістр даних
1	1	0->1->0	Читання з регістра даних
0	1	0->1->0	Читання індикатора зайнятості (DB7) і лічильника адреси (DB0-DB6)

Вибір матриці проводиться один раз в ході процедури ініціалізації РК-модуля, але можна змінити тип матриці і в ході роботи. "Висвічуванням" точки відбувається при подачі на неї одиниці, якщо на точку поданий нуль, то вона "погашена".

В обох типах матриць є додатковий (8-й або 11-й) рядок точок - так званих курсорних. При включенні курсора (програмно) всі 5 точок цього рядка "висвічуються" під тим символом, який буде виведений в наступному циклі виведення на індикацію (курсор - елемент зручності і потрібен не у всіх

пристроях). Сусідні елементи матриці відокремлені один від одного проміжком 0,7 мм. Найчастіше застосовується матриця 5×7 точок, вибір символів в цьому випадку більший, ніж при використанні матриці 5×10 точок.

Символи для відображення розміщуються в області незалежної пам'яті знакогенератора. Оскільки це ПЗУ, набір символів користувач змінити не може. Однак крім області незалежної пам'яті знакогенератор має в своєму складі область оперативної пам'яті CG RAM, куди можна записати "сконструйовані" програмістом символи. Розробники залишають програмісту можливість створювати свої символи: до 8-ми символів для матриці 5×7 точок в одній комірці CG RAM або до 4-х символів для матриці 5×10 точок (один символ - в двох осередках CG RAM). Втім, кількість "штатних" символів настільки велика і різноманітна, що їх вистачає з надлишком.

Знакогенератор - це програмований лічильник, який керує роботою звичайної мікросхеми ПЗУ. Числа в двійковому поданні знакогенератора виводяться з осередків ПЗУ запитаних адрес. Таблиця знакогенератора розбита на клітинки, в яких записані символи за вказаними адресами.

Щоб "запустити" режим запису даних, необхідно спочатку виставити "1" на вході RS модуля РКІ (R/W - на корпусі), а після цього сформувати на вході Е модуля РКІ перепад від "1" до "0", який запускає команду. Після зміни на виході Е "1" на "0" вміст з заздалегідь обраної адреси знакогенератора виведеться в дане знакомісце відеопам'яті РК-дисплею.

До роботи відеопам'яті прив'язаний курсор. Така прив'язка при роботі з подібного роду пристроями дуже корисна. Курсор вказує на те знакомісце, в яке буде виведений на індикацію наступний символ. Поточне знакомісце відеопам'яті РК-дисплею визначає лічильник АС - реверсивний лічильник з програмною установкою автоінкременту і автодекременту і з установкою кроку рахунку. Якщо ж по ходу заповнення рядка символами виникає необхідність у збільшенні або зменшенні значення адреси, сформованої лічильником адреси АС, то програмно необхідно сформувати відповідну команду. При цьому адреси в області відеопам'яті не змінюються. Після

виконання команди зсуву зсувається вміст всіх комірок, переміщаючись з комірки в комірку.

4.5.2. Програмне підключення індикатора для відображення та виведення інформації

Далі налаштовуємо Arduino для взаємодії з дисплеєм. Для цього підключимо стандартну бібліотеку, щоб використовувати дисплей "LiquidCrystal.h". Бібліотека LiquidCrystal входить до стандартного програмного пакету Arduino і дозволяє керувати РК-дисплеєм, що працює на контролері Hitachi HD44780 (або сумісному чіпсеті), що входить до більшості символічних РК-індикаторів. Бібліотека LiquidCrystal працює в 4-х або 8-бітному режимі (на додачу до RS, ENABLE та додаткових ліній керування RW використовуються 4 або 8 рядків даних). Вона містить функції для переміщення курсора на дисплеї, відображення даних, створення символів та очищення дисплея. Описані далі функції реалізовані в бібліотеці.

Функція LiquidCrystal() створює змінну типу LiquidCrystal. Можна використовувати 4 або 8 ліній даних для управління індикатором. Щоб підключити 4 контакти, необхідно пропустити контакти D0 до D3 і залишити їх без зв'язку. Контакт RW може бути заземлений замість контакту плати Arduino. Якщо це так у схемі, необхідно пропустити цей параметр функції. Синтаксис функції LiquidCrystal(): [8]

LiquidCrystal(rs, enable, d4, d5, d6, d7)

LiquidCrystal(rs, rw, enable, d0, d1, d2, d3, d4, d5, d6, d7)

LiquidCrystal(rs, rw, enable, d4, d5, d6, d7)

LiquidCrystal(rs, enable, d0, d1, d2, d3, d4, d5, d6, d7)

Параметри:

- rs - пін плати Arduino, який підключений до RS-контакту РК-дисплея;
- rw - пін плати Arduino, який підключений до контакту RW РК-дисплея (додатково);

- enable - пін плати Arduino, підключений до контакту ENABLE РК-дисплея;
- d0, d1, d2, d3, d4, d5, d6, d7 - це пін плати Arduino, і вони підключені через контакти даних РК-дисплея.
- D0, D1, D2 і D3 є необов'язковими - якщо вони не активовані, РК-дисплеєм можна керувати лише через 4 лінії передачі даних (D4, D5, D6, D7).

Функція `begin ()` визначає розмір індикатора (кількість символів у ширину та висоту). Синтаксис функції `begin ()`: [8]

```
lcd.begin(cols, rows)
```

Параметри:

lcd - змінна типу `LiquidCrystal`;

cols - кількість символів у рядку;

rows - рядки.

Функція `clear ()` очищає РК-екран і переміщує курсор у верхній лівий кут. Вона використовується перед відображенням нової інформації. Під час написання коду виявляється, що її слід використовувати лише для тих частин коду, які виконуються за певних подій (наприклад, після натискання кнопки), інакше можна отримати блимання внаслідок стирання та запису нової інформації; [8]

Синтаксис функції `clear()`:

```
lcd.clear();
```

Параметр: lcd - змінна типу `LiquidCrystal`.

Функція `home()` розміщує курсор у верхньому лівому куті РК-дисплею. Під час відображення послідовного тексту на екрані індикатора вона використовується для визначення початкової позиції. Щоб одночасно очистити екран індикатора необхідно використати `clear()` замість неї. [8]

Синтаксис функції `home()`:

```
lcd.home()
```

Параметр: lcd - змінна типу `LiquidCrystal`.

Функція `setCursor()` використовується для позиціонування курсора РК-дисплею, тобто встановлення положення, де наступний текст буде

відображатися на його екрані. Встановлює курсор на точку з координатами x і y , для цього x може приймати значення від 0 до 15, а y може дорівнювати лише 0 і 1. Синтаксис функції `setCursor()`: [8]

```
lcd.setCursor(col, row)
```

Параметри: `lcd` – змінна типу `LiquidCrystal`; `col` – номер колонки в рядку (для першого знакомісця значення 0); `row` – номер рядка (для першого рядка 0). [8]

Функція `write()` записує символи в РК-дисплей. Синтаксис функції `write()`:

```
lcd.write(data)
```

Параметри: `lcd` – змінна типу `LiquidCrystal`; `data` — символ, який потрібно записати в індикатор.

Функція `createChar()` створює символ, який користувачі можуть використовувати на РК-дисплеї. Підтримує до восьми символів 5×8 пікселів (нумеровані від 0 до 7). Створення кожного символу визначається восьмибайтовим масивом - один байт на рядок. П'ять молодших бітів кожного байта ідентифікують пікселі в рядку. Щоб відобразити власні символи на екрані, необхідно використати функцію `write()` із номером символу як параметр. Синтаксис функції `createChar()`: [8]

```
lcd.createChar (num, data)
```

Параметри: `lcd` - змінна типу `LiquidCrystal`; `num` - номер створеного символу (від 0 до 7); `data` - дані пікселів символів.

Далі використаємо такий функціональний код [8] для підключення дисплея: [8]

```
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
```

Ми налаштували підключення RS, E, DB4-DB7 контактів до цифрових виходів Arduino. Далі потрібно налаштувати роботу дисплею за допомогою команди `lcd.begin(16,2)` на початку програми в тілі функції `setup()`. Потім, використовуючи наведені вище функції, можна відобразити дані на екрані.

4.5.3. Символ градуса для дисплею

Коли пристрій працює, нам потрібно відобразити піктограму градуса, але цієї піктограми на цьому дисплеї немає, тому її потрібно створити. Пам'ять дисплея може містити до восьми символів. Кожен символ відображається на матриці 5x8 балів. Щоб створити символ градуса, вам потрібно вказати, які точки потрібно виділити. У таблиці 4.3 виділено сірі області, в які потрібно записати відповідний стан точок.

Таблиця 4.3 – Відображення стану пікселів символу градуса

0	0	1	1	1
0	0	1	0	1
0	0	1	1	1
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

Далі потрібно написати бітову маску зі створених символів і призначити отриману маску змінній бітового типу. Код для виконання цієї операції такий::

```
byte tempCel[8] =
{
    0b00111,
    0b00101,
    0b00111,
    0b00000,
    0b00000,
    0b00000,
    0b00000,
    0b00000
}
```

```
};
```

Створений символ створимо в дисплей наступним кодом:

```
lcd.createChar(1, tempCel);
```

У цьому записі число 1 представляє кількість символів, що зберігаються в пам'яті, і можливі значення від 1 до 7. Тепер символ градуса можна показати на екрані кодом "\1".

4.5.4. Налаштування підсвічування

Контакти підсвічування дисплея можна підключити до джерела живлення, але він завжди буде споживати енергію і привертати увагу, тому ним потрібно керувати. Для цього необхідно розробити відповідний код. Структурно нам потрібно реалізувати функцію освітлення при натисканні будь-якої кнопки, вимкнути освітлення, коли немає дії протягом встановленого часу, і продовжувати випромінювати світло в наступний інтервал часу, коли буде виконана будь-яка дія.

Увімкнемо підсвічування таким кодом:

```
if ((shortKeyPressedUp || longKeyPressedUp || shortKeyPressedDown ||
longKeyPressedDown) && !enableLight) {
    digitalWrite(LCD_LIGHT_PIN, 1);
    timerOfLighting = millis();
    longKeyPressedUp = false;
    shortKeyPressedUp = false;
    longKeyPressedDown = false;
    shortKeyPressedDown = false;
    enableLight = true;
}
```

Для активації підсвітки достатньо натиснути будь-яку кнопку. Далі на пін підсвідки виводиться логічна одиниця. Усі інші дії при натисканні кнопок

будуть проігноровані, а отже, перший клік не спричинить жодної операції в програмі, а лише запалить підсвічування.

Якщо протягом часу `durationOfLighting` не натиснута жодна клавіша, наступний код вимкне підсвічування:

```
if ( ((millis() - timerOfLighting)/1000) > durationOfLighting) {
    digitalWrite(LCD_LIGHT_PIN, 0);
    enableLight = false;
}
```

Цей код дозволяє визначити тривалість роботи підсвітки з часу останнього її ввімкнення. Тривалість підсвічування можна встановити, записавши необхідний час (у секундах) у змінну `durationOfLighting`. Далі, подібно до попереднього кодового блоку, у вихід контакту підсвічування записується логічний нуль, а у змінну, що відповідає за стан підсвічування, записується логічне заперечення.

Щоб уникнути незручностей при використанні пристрою, час підсвідки повинен бути продовжений під час будь-якої операції:

```
if ((shortKeyPressedUp || longKeyPressedUp || shortKeyPressedDown ||
longKeyPressedDown)) {
    timerOfLighting = millis();
}
```

Вищезазначений код запише новий час до змінної `lightTimer`, і час підсвічування буде розраховуватись відповідно до цього нового значення.

4.6. Підключення та управління випарником та вентилятором

Для управління вентилятором та випарником використовуємо ключі на МОН транзисторах. Ми не використовуємо біполярні, оскільки вони споживають більше енергії, ніж МОН, що додає навантаження порту Arduino, тому вони не мають переваг в даному випадку. Польовий транзистор повинен

бути обраний таким чином, щоб він витримував напругу стік-витік 24 В, струм не менше 1 А, а керуюча напруга 5 В. Виберемо n-канальний IRF520. Він має наступні характеристики: максимальна напруга стоку-джерела становить 100В, максимальна сила струму джерела стоку - 6,5А, максимальна напруга затвора - 20 В, а розсіювання потужності до 50 Вт [30]. Він має кращі характеристики, ніж вимагає завдання, але його використання є розумним через низьку вартість і популярність. Для спрощення схеми такі транзистори застосуємо як перемикачі для вентилятора та випарника.

Транзисторний затвор безпосередньо підключений до цифрового виходу Arduino. Затвор транзистора має певну ємність, що впливає на його роботу та роботу інших компонентів схеми. У цьому випадку зарядка та розрядка буде здійснюватися через порт Arduino, ми не можемо впливати на зарядку, але можемо зменшити навантаження під час розряду. З цією метою затвор кожного транзистора заземлений через резистор 1kОм. Метою вибору цього номіналу є зниження навантаження мікросхеми як мінімум, коли вихід порту увімкнено на високому рівні, одночасно забезпечуючи швидке зменшення ємності та одночасно забезпечуючи швидкий розряд ємності.

Транзистор Q1 для управління випарником, а Q2 - для управління вентилятором. Q1 підключений до цифрового контакту D11, а Q2 - до D10. У коді програми відповідний контакт реєструється як вихідний:

```
pinMode(OUT_EVAP_PIN, OUTPUT);
digitalWrite(OUT_EVAP_PIN, 0);
pinMode(OUT_COOLER_PIN, OUTPUT);
digitalWrite(OUT_COOLER_PIN, 0);
```

У перших двох стрічках вище ми встановлюємо контакт D11 як вихід і застосовуємо логічний 0 до виходу, інакше вихід буде випадковим значенням. Подібним чином у наступних ми чинимо з D10. Далі управління вентилятором зводиться до запису у відповідний контакт високого чи низького рівня. Вмикаємо його наступним чином:

```
digitalWrite (OUT_COOLER_PIN, 1);
```

Метод вимкнення подібний, замість високого рівня записується низький. Управління випарником дещо відрізняється, оскільки для регулювання потужності необхідно використовувати ШІМ. У поточному обладнанні контроль потужності випарника, особливо контроль інтенсивності пароутворення, здійснюється за допомогою наступного кодового блоку:

```
PWMIntensitySteam = map(intensitySteam, 0, 100, 0, 255);
```

```
analogWrite(OUT_EVAP_PIN, PWMIntensitySteam);
```

Далі наводиться опис використовуваних змінних та функцій:

- `intensitySteam` - містить змінну для даної потужності випаровування в діапазоні від 1 до 100%.

- `map(value, a1, a2, b1, b2)` - функція, що використовується для перетворення чисел в діапазоні `a1` в `a2` в числа в діапазоні `b1` до `b2`. У цьому випадку значення змінної `intensitySteam` в діапазоні 0-100 перетворюється на число в діапазоні 0-255, яке використовується в ШІМ.

- `OUT_EVAP_PIN` - Вихід, з якого подається ШІМ-сигнал на навантаження.

- `PWMIntensitySteam` - змінна, яка зберігає значення, яке має бути встановлене на виході за допомогою ШІМ.

4.7. Меню пристрою та код управління

Для полегшення управління пристроєм було розроблено меню з найменшою кількістю налаштувань та найбільшим обсягом інформації. Структурна схема показана на рисунку 4.3. У головному меню (рис. 4.4) короткочасно натисніть клавішу, щоб перемістити курсор до потрібного пункту. Натисніть та утримуйте більше 1 секунди, щоб підтвердити вибір.

У режимі зволоження (рис. 4.5) на екрані відображається поточна вологість після "`hC =`", цільова вологість після "`hF =`" та температура за Цельсієм після "`t =`". Ви можете збільшити або зменшити необхідну вологість з кроком 2%, просто натиснувши відповідну кнопку, і в такому випадку зміни

можна вносити в режимі реального часу. Ви можете вийти з цього режиму та будь-якого іншого режиму, довго натискаючи кнопку "вниз".

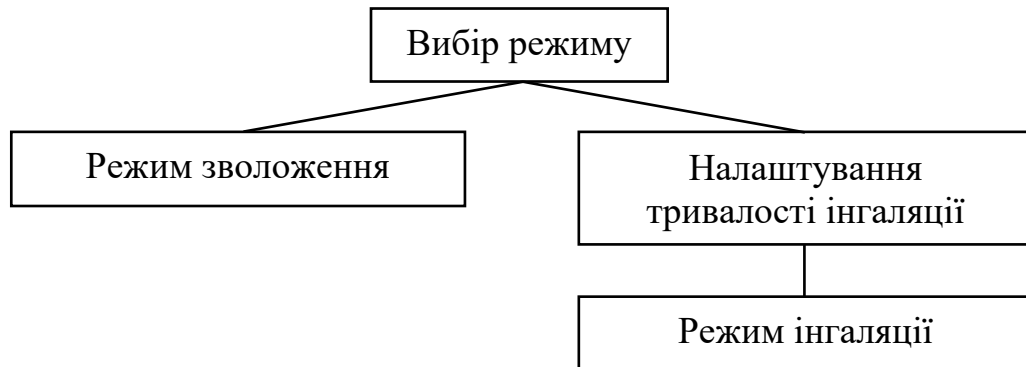


Рисунок 4.3 – Блок-схема меню

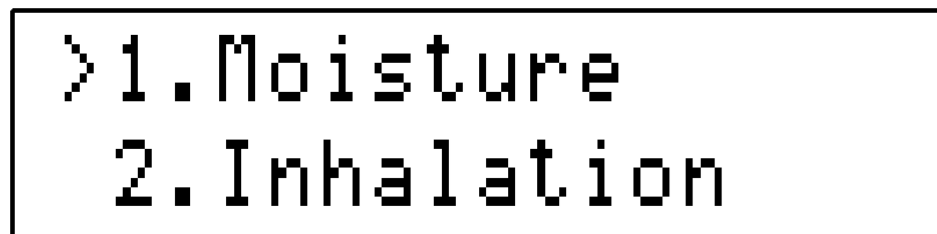


Рисунок 4.4 – Головне меню

У меню для встановлення тривалості інгалювання (рис. 4.6) ви можете встановити бажану тривалість, крок і мінімальна тривалість становлять 1 хвилину. Ви можете підтвердити час тривалим натисканням кнопки "вгору".

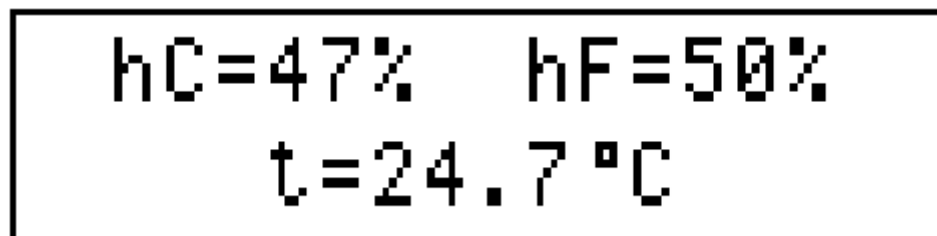


Рисунок. 4.5 – Режим зволоження

Після цього, згідно з блок-схемою, переходимо у наступне меню (рис. 4.7). У цьому режимі на екрані відображається час, що пройшов, після літери

"P", час, що залишився, після літери "L", поточна температура води після "t =" та відсоток потужності після тексту "POW =". Останній параметр можна змінити, короткочасно натискаючи кнопку з кроком 5% в діапазоні від 0 до 100%.



Рисунок. 4.6 – Режим налаштування інгаляції

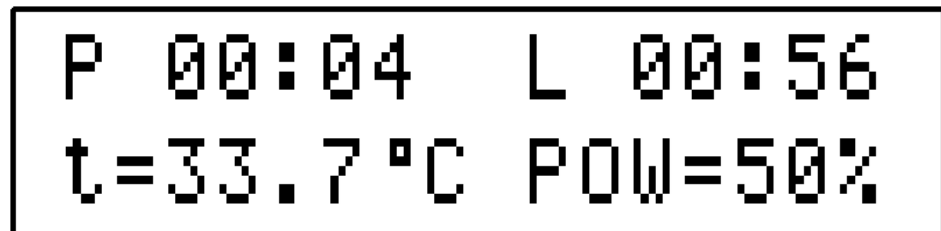


Рисунок. 4.7 – Режим інгаляції

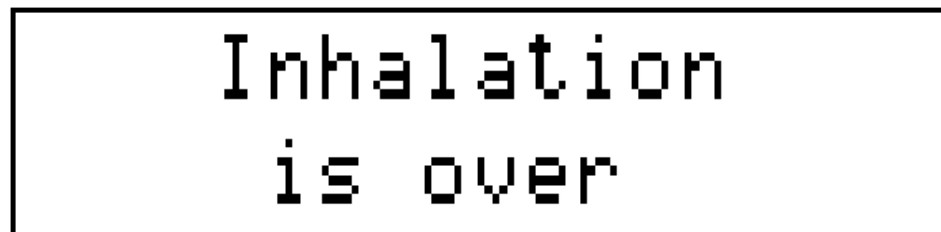


Рисунок. 4.8 – Сповіщення про зупинку інгаляції

Після закінчення встановленого часу на екрані відображатиметься інформація про закінчення вдиху (рис. 4.8). Також в режимі інгалювання є також контроль температури води. Якщо максимальна або мінімальна межа перевищена, робота припиняється і відображається попередження про перевищення допустимого обмеження. Відповідні зображення на екрані показані на малюнках 4.9 та 4.10.



Рисунок. 4.9 – Попередження про холодну воду



Рисунок. 4.10 – Попередження про гарячу воду

4.8. Розробка живлення

У розробленому рішенні прилад має три напруги живлення: 5 В, 12 В і 24 В. Для роботи кожного компонента така напруга повинна подаватись до них з найменшим відхиленням, інакше компоненти працюватимуть нестабільно або будуть несправні. Тому необхідно розробити блок живлення, який буде забезпечувати всі три напруги на виході.

Схема джерела живлення має наступні варіанти: комбінація джерела живлення з однією вихідною напругою та цифровим перетворювачем або джерело, яке видає усі три напруги на виході.

Розглянемо блок живлення з трьома вихідними напругами. Блок живлення має трансформатор і схему для перетворення змінного струму в постійний. Більше того, оскільки трансформатор має три вторинні обмотки з різною вихідною напругою, потрібні три фільтри. Це ускладнить прилад.

Коли джерело живлення з вихідною напругою використовується в поєднанні з цифровим перетворювачем, воно також може використовуватися в розробленому пристрої. Якщо потужність обладнання, що живиться від високої напруги, низька, рекомендується використовувати його, інакше

будуть відносно великі втрати тепла. У нашому випадку цифрова частина схеми споживає найменше енергії, коли напруга джерела живлення становить 5 В, тоді як вентилятор 12 В споживає трохи більше енергії, а випарник споживає найбільше енергії, коли напруга живлення становить 24 В. З вищевказаних причин недоцільно використовувати такий блок живлення.

Блок живлення з вихідною напругою та цифровим перетворювачем є найбільш придатним для живлення обладнання. Для цього використаємо блок живлення з вихідною напругою 24В.

Випарник буде живитись безпосередньо від джерела живлення. Для того, щоб живити вентилятор з двигуном 12 В, для зменшення напруги потрібно використовувати цифровий перетворювач постійного струму. Для цього використаємо чіп L7812. Він реалізує перетворювач постійного струму 12 В з двома зовнішніми конденсаторами. Мікросхема має три контакти: V_{in} , V_{out} , GND-вхід, вихід і заземлення відповідно. Вхідна напруга може знаходитися в межах від 15 до 35 В. У нашому випадку вона дорівнює 24 В. Отже, мікросхема буде працювати з допустимим характеристичним запасом. Вихідний струм може бути до 1,5 А, достатній для живлення цифрової частини схеми та невеликого вентилятора. Коли відбувається перевантаження, мікросхема має вбудовану схему для запобігання перегріванню та перевищенню максимального вихідного струму, що вимкне вихід, поки нормальна робота не буде відновлена. Струм, споживаний вентилятором, становить 0,08 А, тому наступний перетворювач напруги та цифрова частина схеми можуть споживати до 1,42 А [31].

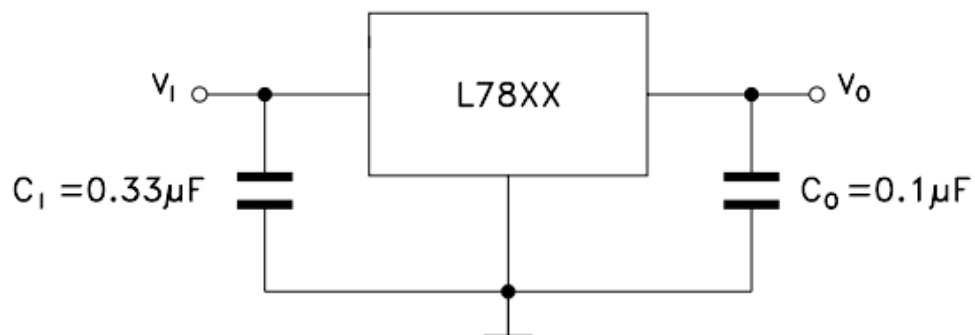


Рисунок 4.11 – Підключення мікросхем L78XX [31]

Напруга живлення цифрової частини становить 5 В, тому для її живлення потрібно використовувати іншу цифрову схему зниження постійного струму, вихідна напруга якої становить 5 В. Виберемо чіп L7805. Він має всі подібні характеристики та схеми захисту. Він також має подібну схему підключення[31]. Для живлення ланцюга необхідно використовувати джерело живлення з вихідною напругою 24 В і потужністю не менше 25 Вт.

4.1. Висновок до поточного розділу

Поточна частина описує принцип підключення всіх компонентів, що використовуються в Arduino, особливо їх фізичну схему підключення та код, що дозволяє використовувати відповідні компоненти. Створено блок-схему, що відображає структуру меню та результат реалізації. На основі вищезазначених даних була розроблена принципова схема пристрою, яка наведена у Додатку 1, і розроблений до неї код повинен бути записаний в Arduino. Блок-схема приладу наведена на рисунку 4.12. Код програми наведено в репозиторії GitHub. Режим доступу до ресурсу: https://github.com/YuriiYurchyk/diploma_mag/blob/master/code.c

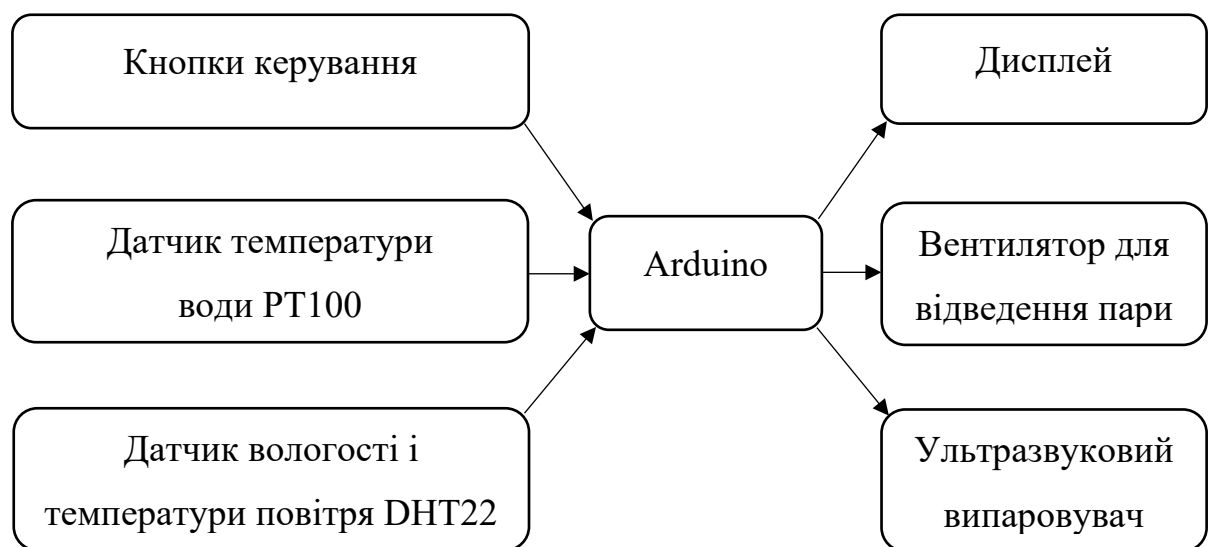


Рисунок 4.12 – Блок-схема приладу

Програма відтворена в середовищі моделювання електронних схем Proteus Design 8. Код програми скомпільований та завантажений на плату, яка моделює Arduino Nano. Схема була успішно протестована і створений прототип.

5. Стартап-проект

5.1. Назва проекту

«Ультразвуковий розпилювач рідини для контролю вологості та інгаляції»

5.2. Короткий опис проекту

Розроблено прилад, який суміщує в собі функції кімнатного зволожувача повітря з регулюванням рівня вологості та інгалятора з можливістю регулювання інтенсивності випарювання рідини та контролем за температурою

Застосування розробленого приладу вирішує наступні задачі:

- контроль за вологістю повітря у житлових приміщеннях;
- інгаляція;

Розроблений прилад дозволяє на оптимальному рівні підтримувати вологість у приміщеннях з пониженою вологістю в теплу пору року та актуальний до використання у всіх будинках зимою у зв'язку з пониженням вологості повітря при нагріванні повітря з вулиці у приміщеннях.

За видом проект відноситься до проектів «новий продукт на існуючий ринок».

Споживчий сегмент – B2C.

5.3. Бізнес-модель

5.3.1. Цінний продукт

Продукт передбачає контроль вологості як основну функцію і, за потреби, може бути використаний для інгаляції.

5.3.2. Сегмент споживачів

Продукт націлений на широкий ринок споживачі B2C. Зокрема покупцями можуть бути домогосподарки, молоді сім'ї, оскільки для малих дітей особливо важливе підтримання оптимального рівня вологості, крім того для малих дітей інгаляція часто найоптимальніший спосіб доставки ліків.

5.3.3. Канали збуту

Без участі посередників використовуються прямі канали збуту, пов'язані з доставкою товарів та послуг. Вони забезпечують доступ до кінцевих споживачів, що забезпечує значні переваги, такі як можливість збирати маркетингову інформацію та прямий вплив на споживачів, але вимагає великих фінансових ресурсів. Прямі канали в основному використовуються виробниками, які контролюють свої маркетингові плани та діють на обмеженому цільовому ринку. Зазвичай виробники зацікавлені продавати свою продукцію безпосередньо споживачам з власних регіональних складів.

Діяльність збуту промислових підприємств із використанням прямих продажів може здійснюватися через оптову базу збуту виробника, склад та оптовий офіс. Можна продавати через Інтернет-ресурси.

5.3.4. Взаємодія зі споживачами

Сформувати лояльну споживчу базу, яка створить досвід споживання продукції та підтримуватиме її на високому рівні, зосередившись на сферах, які справді важливі для споживачів. Завод не лише прагне збільшити участь співробітників компанії, але й створив внутрішню культуру, що досягає високих показників, яка не тільки забезпечує набір стандартних продуктів та технологій, але й сприяє подальшому розвитку компанії.

5.3.5. Прибуток (монетизація)

Отримання прибутку від продажу готових приладів.

5.3.6. Ключові види діяльності

1. Наукова діяльність - це діяльність з інтелектуального створення, спрямована на здобуття та використання нових знань. Основними його формами є фундаментальні та прикладні дослідження.

2. Виробництво готових приладів – певний промисловий процес створення з окремих компонентів готових виробів.

3. Маркетинг - це творча управлінська діяльність, завданням якої є розвиток продуктів, послуг та ринків праці шляхом оцінки потреб споживачів та вжиття практичних заходів для їх задоволення.

5.3.7. Ключові ресурси

Матеріальні – електронні компоненти, пластикові корпуси. Технологія виробництва. Захисні документи (патенти). Працівники наукової та технічної сфери.

5.3.8. Ключові партнери

Постачальники сировини для виробництва: ТОВ «KLONA», ТОВ «Imrad».

Постачальник електроенергії – ПАТ «Київенерго».

Підприємство з можливістю оренди потужностей виробництва: завод виробництва радіоелектроніки.

Маркетингові компанії – ТОВ «SEO – studio».

Логістика та ринкові партнери: просування сайтів, оптимізація «SEOOWL», «DELPOST».

5.3.9. Витрати

- вартість оренди промислових потужностей;
- закупівля матеріалів;
- доставка матеріалів;
- електрика;
- пакування готової продукції;
- зарплата;
- логістика;
- маркетинг;
- підтримка Інтернет-ресурсів.

5.3.10. Споживчі властивості товару

Виріб має тривалий термін служби, тим самим вирішуючи споживчі проблеми, пов'язані з витратами на обслуговування, а також має кращі експлуатаційні та економічні показники. Порівняно з існуючими на ринку товарами, мінливість сфери випуску продукції може значно розширити сферу її використання в різних галузях та зробити її більш конкурентоспроможною.

5.3.11. Дослідження ринку

На основі результатів аналізу існуючого товарного ринку для подібних цілей ми можемо зробити наступні висновки:

- приладі для інгаляції є дорогими;
- відсутні прилади, які суміщують дві функції: інгаляції та зволоження.

5.3.12. Дослідження конкурентного оточення

Продукцію подібного призначення виробляють українські та іноземні компанії. Аналіз показав продукцію України, країн СНД та Китаю як найбільш вірогідних конкурентів. В даний час переваги нашої продукції з точки зору якості та економічності дозволяють їм посідати місце на ринку подібних товарів. Можливими конкурентами є обладнання китайського виробництва, яке може конкурувати за ціною.

5.3.13. Маркетингова стратегія просування

Маркетингова стратегія просування проекту включатиме:

- просування проекту в Інтернеті;
- участь у галузевих виставках та конференціях;
- пропозиції потенційним покупцям;
- зустрічі безпосередньо з покупцями для демонстрації та продуктів;
- поступовий вихід на український ринок, потім на міжнародний ринок.

5.4. Елементи фінансового плану

5.4.1. Опис бізнес-проекту

Мета проекту – отримання прибутку шляхом продажу розроблених виробів.

Актуальність проекту – існує проблема низької вологості в житлових приміщеннях в зимову пору року і меншою мірою в літню та вартість приладів для інгаляції в даний момент висока.

6. Висновки

У цій роботі розглядається можлива реалізація кожного блоку пристрою та відбирається найкраще обладнання відповідно до призначення та умов роботи кожного компонента. Важливим критерієм при виборі компонентів є мінімальне задоволення вимог до використання. Це важливо для ефективного використання ресурсів. ШІМ використовується для управління потужністю. Розглянуто виконання цього методу в Arduino та синтаксис програмного використання цієї функції, що дозволяє вмикати ШІМ на певному виході контролера. На основі обраних компонентів була розроблена принципова схема пристрою, який був розроблений для підключення до джерела живлення постійного струму 24 В. Розроблені вимоги джерела живлення, необхідного для роботи обладнання.

Схема розроблена у середовищі Proteus Design 8. В процесі моделювання були виправлені можливі недоліки. Оскільки пристрій був успішно змодельований програмно, то це говорить про працездатність приладу.

7. Перелік використаної літератури

1. Perry, R.H. Perry's Chemical Engineers' Handbook, Section 12-3, Psychrometry / R.H. Perry, D.W. Green // McGraw-Hill. – 2007. – 540 с. –
Режим доступу до ресурсу:
https://www.academia.edu/34738909/Perrys_Chemical_Engineers_Handbook_8thEd
2. Climate – Humidity indexes // Encyclopaedia Britannica. –
Режим доступу до ресурсу: <https://www.britannica.com/science/climate-meteorology/Humidity-indexes>
3. Climate/humidity table. Transport Information Service of the German Insurance Association. –
Режим доступу до ресурсу: http://www.tis-gdv.de/tis_e/misc/klima.htm
"Optimum Humidity Levels for Home. –
Режим доступу до ресурсу:
<https://www.airbetter.org/optimum-humidity-levels-home>
4. Effect of Humidity and Condensation on Power Electronics Systems. –
Режим доступу до ресурсу: <https://www.semikron.com/dl/service-support/downloads/download/semikron-application-note-effect-of-humidity-and-condensation-on-power-electronics-systems-en-2016-07-15-rev-00>
5. Чучалин, А.Г. Место небулайзеров в ингаляционной терапии хронических обструктивных заболеваний легких / А.Г. Чучалин, Н.П. Княжеская, М.О. Потапова. // Русский медицинский журнал. – 2006. – №7 – 521–524 с.
6. Arduino – Products. –
Режим доступу до ресурсу: <https://www.arduino.cc/en/Main/Products>
7. Петин, В.А. Проекты с использованием контроллера Arduino / В.А. Петин. – Санкт-Петербург. – 2014. – 20–66 с.
8. Виглеб, Г. Датчики. Устройство и применение / Г. Виглеб. – Москва: «Мир». – 1989. – 14–22 с.

9. Шашков, А.Г. Динамические свойства цепей с термисторами / А.Г. Шашков, А.С. Касперович – М.: Госэнергоиздат – 1962. – 210 с.
10. Терморезистивные датчики. Датчики и сенсоры онлайн журнал. – Режим доступа до ресурсу: <http://datchikisensor.narod.ru/0128.html>
11. Тумайкин, Д.С. Прецизионный термометр для промышленного применения на основе термочувствительных кварцевых резонаторов. / Д.С. Тумайкин, М.С. Тумайкин // Компоненты и технологии. – 2008. – №1. – 15–16 с.
12. Дивин, А. Г. Методы и средства измерений, испытаний и контроля. Часть 3. / А. Г. Дивин, С. В. Пономарев. – Тамбов: ФГБОУ ВПО «ТГТУ». – 2013. – 31–32 с.
13. Umidificatore a caldo Respirasano. – Режим доступа до ресурсу: http://www.chicco.by/manuals/staroe/User_Manual_Healthy_Breathing.pdf
14. Ultrasonic Humidifiers. – Режим доступа до ресурсу: <https://www.cedengineering.com/userfiles/Ultrasonic%20Humidifiers.pdf>
15. Ивченко Ю.А., Федоров А.А. Чем измерить влажность? // Датчики и системы. – 2003. – №8. – 53–54 с.
16. Кривченко, И.В. AVR – микроконтроллеры: очередной этап на пути развития / И.В. Кривченко // Компоненты и технологии. – 2002. – №3. – 47–60 с.
17. Пархоменко, Д. А., Смирнов, Є. М. Розробка радіоелектронних схем на основі мікроконтролерів (на прикладі AVR мікроконтролерів фірми Atmel) / Д. А. Пархоменко, Є. М. Смирнов. – Київ: видавництво радіофізичного факультету Київського національного університету імені Тараса Шевченка – 2013. – 110–120 с.
18. Мысловский Э.К. Краткий обзор популярных семейств современных микроконтроллеров / Э.К. Мысловский, А.С. Власов, М.Д. Акристиний // Электронные компоненты. – 2002. – №5. – 22–40 с.

19. Ахметов М.К. 8-разрядные RISC микроконтроллеры / М.К. Ахметов // Новости о микросхемах. –
Режим доступа до ресурсу: http://chipnews.gaw.ru/html.cgi/arhiv/99_09/stat_2.htm
20. Бугаев, В.И. Лабораторный практикум для изучения микроконтроллеров архитектуры ARM Cortex-M4 на базе отладочного модуля STM32F4 Discovery / В.И. Бугаев, М.П. Мусиенко, Я.М. Крайнык. – Москва-Николаев: МФТИ-ЧГУ. – 2013. – 71–75 с.
21. STMicroelectronics To Acquire Genesis Microchip. –
Режим доступа до ресурсу: <https://investors.st.com/news-releases/news-release-details/stmicroelectronics-acquire-genesis-microchip>
22. STM32F103x8 Datasheet. –
Режим доступа до ресурсу: <https://www.st.com/resource/en/datasheet/cd00161566.pdf>
23. STM32F100R4 Datasheet. –
Режим доступа до ресурсу: <https://datasheetspdf.com/datasheet/STM32F100R4.html>
24. Analog Pulse Width Modulation. // Texas Instruments. –
Режим доступа до ресурсу: <http://www.ti.com/lit/ug/slau508/slau508.pdf>
25. RC1602B-datasheet. –
Режим доступа до ресурсу: https://www.raystar-optronics.com/upload_files/monochrome-lcd-module/16x2-character-lcd-display/RC1602B-datasheet.pdf
26. Mini Thermocouple (Thermal Resistance). PT100 Datasheet. –
Режим доступа до ресурсу: <http://www.wr-wz.com/common/down/name/580d5472a1a32.pdf.html>
27. Digital-output relative humidity & temperature sensor/module DHT22 (DHT22 also named as AM2302). –
Режим доступа до ресурсу: <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>
28. EB40101S2-000U-999 Datasheet. –

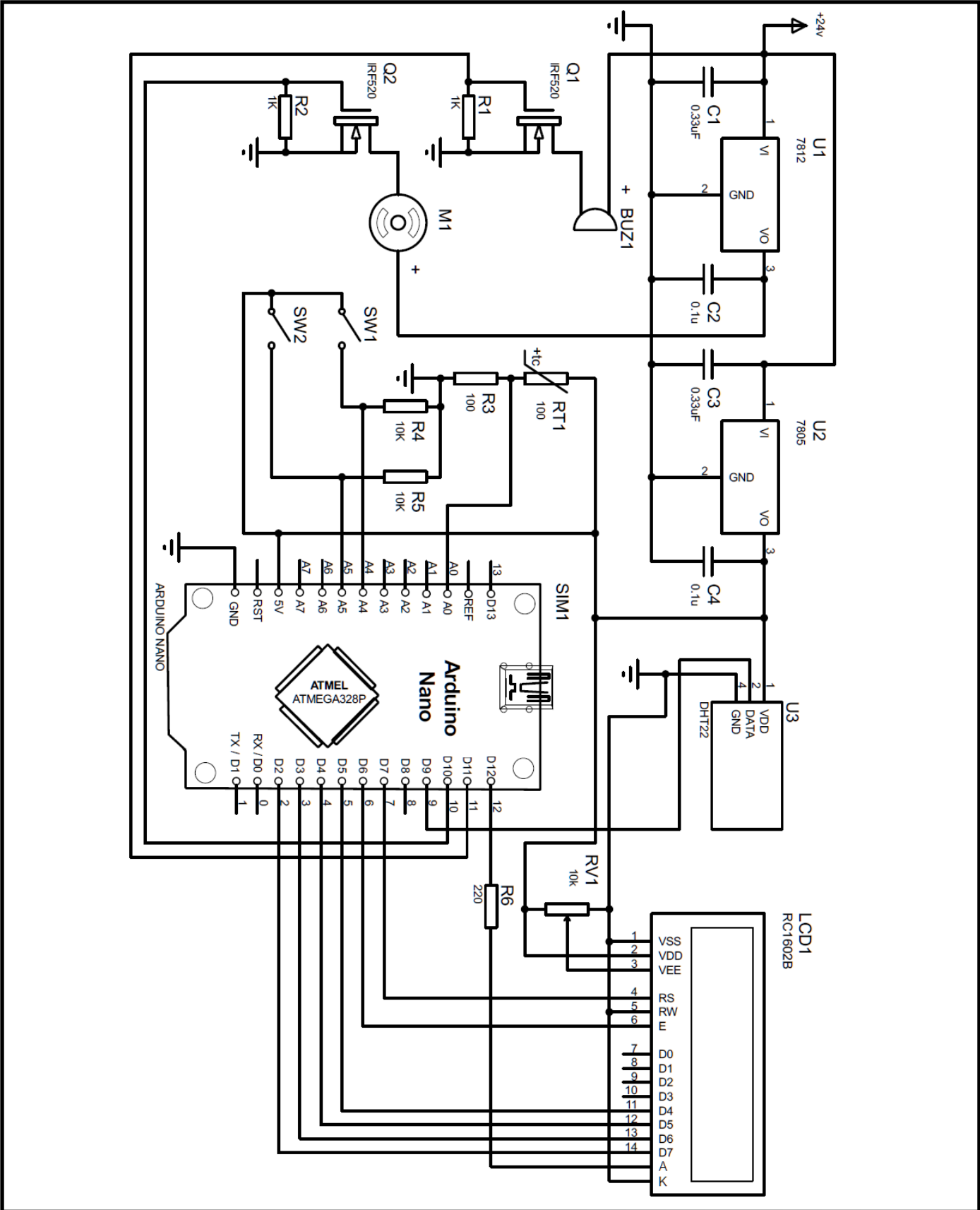
Режим доступа до ресурсу: <https://www.jameco.com/Jameco/Products/ProdDS/2167234.pdf>

29. Power MOSFET IRF520, SiHF520 Datasheet. –

Режим доступа до ресурсу: <https://www.vishay.com/docs/91017/91017.pdf>

30. Positive voltage regulators L7800 series. –

Режим доступа до ресурсу: <https://www.jameco.com/Jameco/Products/ProdDS/889305.pdf>



					ДП92.9210.002.001.ПС		
Змн.	Арк.	№ докум.	Підпис	Дата	Принципова схема		
Розроб.	Юрчик Ю.І.						
Перевір.	Якименко Ю.І.						
Реценз.							
Н. Контр.							
Затверд.					Лім. Арк. Акрушіє		
					75 75		
					НТУУ «КПІ ім. І.Сікорського»		

